



UNIVERSIDAD
NACIONAL
DE COLOMBIA

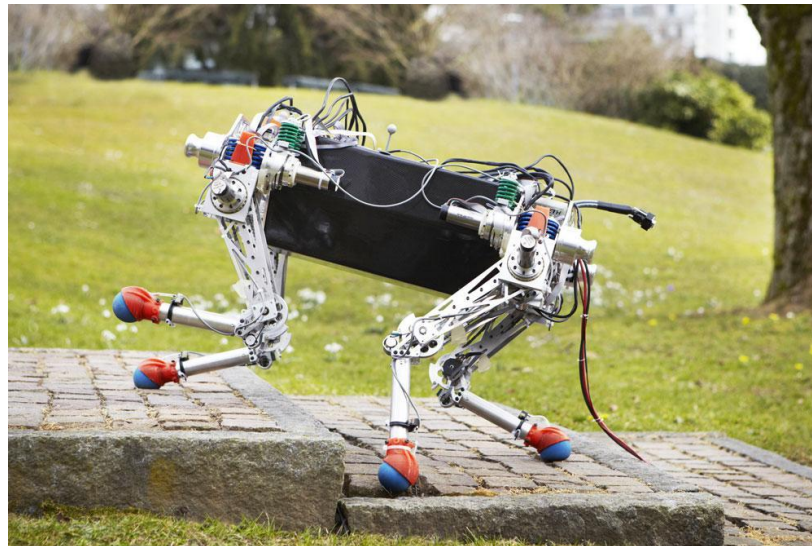
Modelamiento de Legged Robots con ROS

Seykarin Mestre Muelas

Facultad de minas - Sede Medellín

Introducción

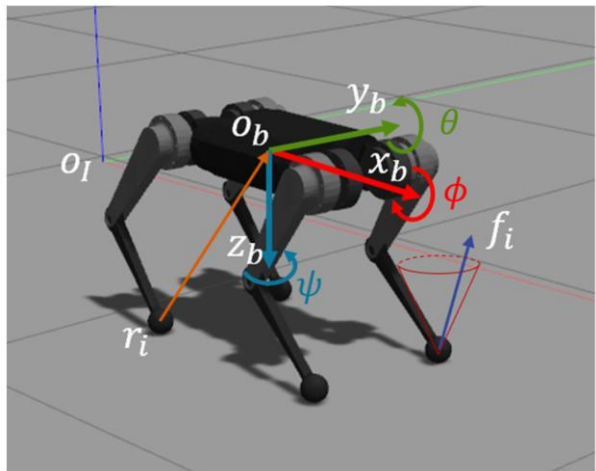
- Los *legged robots* ofrecen versatilidad y estabilidad en terrenos irregulares.
- ROS permite el desarrollo y simulación de estos sistemas de manera modular.
- Esta presentación aborda los principios matemáticos detrás del modelamiento de *legged robots* basándose en la tesis Optimization-based motion planning for legged robots



Problemas:

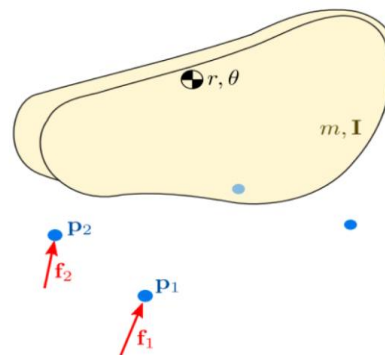
- Cambio de las ecuaciones que rigen el proceso, aumentando notablemente su complejidad.
- Introducción a un nuevo entorno de trabajo.
- Problemas de hardware a la hora de simular.

$$M(q)\ddot{q} + h(q, \dot{q}) = S^T \tau + J(q)^T f$$



$$\frac{d}{dt} A \dot{q} = \begin{bmatrix} mg + \sum_{i=1}^{n_i} f_i \\ \sum_{i=1}^{n_i} f_i \times (r(q) - p(q)) \end{bmatrix}$$

$$A(q)\dot{q} + \dot{A}(q)\dot{q} = \begin{bmatrix} mg + \sum_{i=1}^{n_i} f_i \\ \sum_{i=1}^{n_i} f_i \times (r(q) - p(q)) \end{bmatrix}$$



Herramientas



- Búsqueda en el navegador
- Inteligencias artificiales.
- Asesorías



Modelos

Mejora del entendimiento de modelos matemáticamente avanzados o desconocidos.

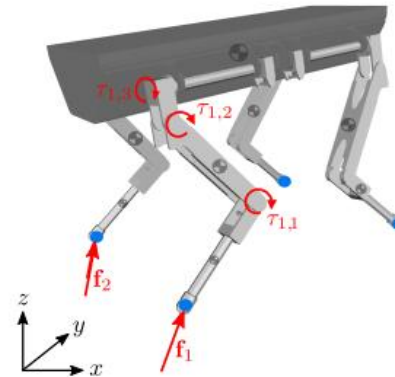
- Cuerpo rígido dinámico

$$M(q)\ddot{q} + h(q, \dot{q}) = S^T \tau + J(q)^T f$$

- Dinámica centroidal

$$\frac{d}{dt} A \dot{q} = \begin{bmatrix} mg + \sum_{i=1}^{n_i} f_i \\ \sum_{i=1}^{n_i} f_i \times (r(q) - p(q)) \end{bmatrix}$$

Rigid Body Dynamics (RBD)



Modelos

- Cuerpo rígido dinámico único

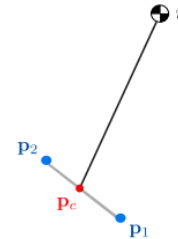
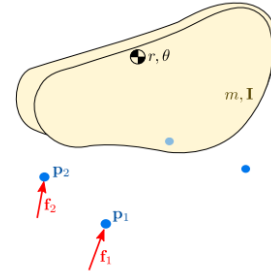
$$m\ddot{r} = mg + \sum_{i=1}^{n_i} f_i$$

$$I(\theta)\dot{w} + w \times I(\theta)w = \sum_{i=1}^{n_i} f_i \times (r - p_i)$$

- Péndulo invertido lineal

$$m\ddot{r}_x = \sum_{i=1}^{n_i} f_{i,x} = \frac{mg}{r_z - p_z} \left(r_x - \frac{\sum_{i=1}^{n_i} f_{i,z} p_{i,x}}{\sum_{i=1}^{n_i} f_{i,z}} \right) = \frac{mg}{h} (r_x - p_{c,x})$$

$$\ddot{r}_c = \frac{g}{z_c} (r_x - p_{c,x})$$



Planes a futuro

- Analizar las formas en las que es controlado este Legged Robot.

- Presentar una explicación simplificada de la programación del robot con el fin de relacionarlo con las ecuaciones presentadas.

```

_setup_util.py - Docker_ROS_Noetic_OCS2 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
  DOCKER_ROS_NOETIC_OCS2
    catkin_ws
      build
        ocs2_arma1_models
        ocs2_arma1_mpc
        ocs2_ballbot
        ocs2_ballbot_mpcnet
        ocs2_ballbot_ros
        ocs2_cartpole
        ocs2_cartpole_ros
        ocs2_centroidal_model
        ocs2_core
        ocs2_ddp
        ocs2_doc
        ocs2_double_integrator
        ocs2_double_integrator_ros
        ocs2_frank_wolfe
        ocs2_ipm
        ocs2_legged_robot
        atomic_configure
        _setup_util.py
        ! rosinstall
        $ env.sh
        $ local_setup.bash
        $ local_setup.sh
        $ local_setup.zsh
        $ setup.bash
        $ setup.sh
        $ setup.zsh
        > bin
        > catkin
        > catkin_generated
        > CMakeFiles
        > gtest
        > include
  _/Docker_ROS_Noetic_OCS2/catkin_ws/build/ocs2_double_integrator_ros/CES:
    221 env_hook_dir="/tmp/ocs2/RH_FOOT_velocity/cppad_generated/RH_FOOT_velocity/libcppadcg_tmp569110.so"
    222 if os.path.isdir(env_hook_dir):
    223     for filename in sorted(os.listdir(env_hook_dir)):
    224         if filename.endswith('.%s' % generic_env_hook_ext):
    225             # remove previous env hook with same name if present
    226             if filename in generic_env_hooks_by_filename:
    227                 i = generic_env_hooks.index(generic_env_hooks_by_filename[filename])
    228                 generic_env_hooks.pop(i)
    229                 generic_env_hooks_workspace.pop(i)
    230
    231     generic_env_hooks_workspace.append(os.path.join(env_hook_dir, filename))
    232
    233     lines.append(comment('found environment hooks in workspaces'))
    234
    235     generic_env_hooks = []
    236     generic_env_hooks_workspace = []
    237     specific_env_hooks = []
    238     specific_env_hooks_workspace = []
    239     generic_env_hooks_by_filename = {}
    240     specific_env_hooks_by_filename = {}
    241     generic_env_hook_ext = 'bat' if IS_WINDOWS else 'sh'
    242     specific_env_hook_ext = environ['CATKIN_SHELL'] if not IS_WINDOWS and 'CATKIN_SHELL' in environ and environ['CATKIN_SHELL'] else None
    243     # remove non-workspace paths
    244     workspaces = [path for path in cmake_prefix_path.split(os.pathsep) if path and os.path.isfile(os.path.join(path, CATKIN_MARKER_FILE))]
    245
    246     for workspace in workspaces:
    247         env_hook_dir = os.path.join(workspace, 'etc', 'catkin', 'profile.d')
    248         if os.path.isdir(env_hook_dir):
    249             for filename in sorted(os.listdir(env_hook_dir)):
    250                 if filename.endswith('.%s' % generic_env_hook_ext):
    251                     # remove previous env hook with same name if present
    252                     if filename in generic_env_hooks_by_filename:
    253                         i = generic_env_hooks.index(generic_env_hooks_by_filename[filename])
    254                         generic_env_hooks.pop(i)
    255                         generic_env_hooks_workspace.pop(i)
    256
    257                     lines.append('source %s' % os.path.join(env_hook_dir, filename))
    258                     generic_env_hooks.append(filename)
    259                     generic_env_hooks_workspace.append(os.path.join(env_hook_dir, filename))
    260                     generic_env_hooks_by_filename[filename] = generic_env_hooks_workspace.pop(i)
    261
    262     lines.append('')
    263
    264     lines.append('')
    265
    266     lines.append('')
    267
    268     lines.append('')
    269
    270     lines.append('')
    271
    272     lines.append('')
    273
    274     lines.append('')
    275
    276     lines.append('')
    277
    278     lines.append('')
    279
    280     lines.append('')
    281
    282     lines.append('')
    283
    284     lines.append('')
    285
    286     lines.append('')
    287
    288     lines.append('')
    289
    290     lines.append('')
    291
    292     lines.append('')
    293
    294     lines.append('')
    295
    296     lines.append('')
    297
    298     lines.append('')
    299
    300     lines.append('')
    301
    302     lines.append('')
    303
    304     lines.append('')
    305
    306     lines.append('')
    307
    308     lines.append('')
    309
    310     lines.append('')
    311
    312     lines.append('')
    313
    314     lines.append('')
    315
    316     lines.append('')
    317
    318     lines.append('')
    319
    320     lines.append('')
    321
    322     lines.append('')
    323
    324     lines.append('')
    325
    326     lines.append('')
    327
    328     lines.append('')
    329
    330     lines.append('')
    331
    332     lines.append('')
    333
    334     lines.append('')
    335
    336     lines.append('')
    337
    338     lines.append('')
    339
    340     lines.append('')
    341
    342     lines.append('')
    343
    344     lines.append('')
    345
    346     lines.append('')
    347
    348     lines.append('')
    349
    350     lines.append('')
    351
    352     lines.append('')
    353
    354     lines.append('')
    355
    356     lines.append('')
    357
    358     lines.append('')
    359
    360     lines.append('')
    361
    362     lines.append('')
    363
    364     lines.append('')
    365
    366     lines.append('')
    367
    368     lines.append('')
    369
    370     lines.append('')
    371
    372     lines.append('')
    373
    374     lines.append('')
    375
    376     lines.append('')
    377
    378     lines.append('')
    379
    380     lines.append('')
    381
    382     lines.append('')
    383
    384     lines.append('')
    385
    386     lines.append('')
    387
    388     lines.append('')
    389
    390     lines.append('')
    391
    392     lines.append('')
    393
    394     lines.append('')
    395
    396     lines.append('')
    397
    398     lines.append('')
    399
    400     lines.append('')
    401
    402     lines.append('')
    403
    404     lines.append('')
    405
    406     lines.append('')
    407
    408     lines.append('')
    409
    410     lines.append('')
    411
    412     lines.append('')
    413
    414     lines.append('')
    415
    416     lines.append('')
    417
    418     lines.append('')
    419
    420     lines.append('')
    421
    422     lines.append('')
    423
    424     lines.append('')
    425
    426     lines.append('')
    427
    428     lines.append('')
    429
    430     lines.append('')
    431
    432     lines.append('')
    433
    434     lines.append('')
    435
    436     lines.append('')
    437
    438     lines.append('')
    439
    440     lines.append('')
    441
    442     lines.append('')
    443
    444     lines.append('')
    445
    446     lines.append('')
    447
    448     lines.append('')
    449
    450     lines.append('')
    451
    452     lines.append('')
    453
    454     lines.append('')
    455
    456     lines.append('')
    457
    458     lines.append('')
    459
    460     lines.append('')
    461
    462     lines.append('')
    463
    464     lines.append('')
    465
    466     lines.append('')
    467
    468     lines.append('')
    469
    470     lines.append('')
    471
    472     lines.append('')
    473
    474     lines.append('')
    475
    476     lines.append('')
    477
    478     lines.append('')
    479
    480     lines.append('')
    481
    482     lines.append('')
    483
    484     lines.append('')
    485
    486     lines.append('')
    487
    488     lines.append('')
    489
    490     lines.append('')
    491
    492     lines.append('')
    493
    494     lines.append('')
    495
    496     lines.append('')
    497
    498     lines.append('')
    499
    500     lines.append('')
    501
    502     lines.append('')
    503
    504     lines.append('')
    505
    506     lines.append('')
    507
    508     lines.append('')
    509
    510     lines.append('')
    511
    512     lines.append('')
    513
    514     lines.append('')
    515
    516     lines.append('')
    517
    518     lines.append('')
    519
    520     lines.append('')
    521
    522     lines.append('')
    523
    524     lines.append('')
    525
    526     lines.append('')
    527
    528     lines.append('')
    529
    530     lines.append('')
    531
    532     lines.append('')
    533
    534     lines.append('')
    535
    536     lines.append('')
    537
    538     lines.append('')
    539
    540     lines.append('')
    541
    542     lines.append('')
    543
    544     lines.append('')
    545
    546     lines.append('')
    547
    548     lines.append('')
    549
    550     lines.append('')
    551
    552     lines.append('')
    553
    554     lines.append('')
    555
    556     lines.append('')
    557
    558     lines.append('')
    559
    560     lines.append('')
    561
    562     lines.append('')
    563
    564     lines.append('')
    565
    566     lines.append('')
    567
    568     lines.append('')
    569
    570     lines.append('')
    571
    572     lines.append('')
    573
    574     lines.append('')
    575
    576     lines.append('')
    577
    578     lines.append('')
    579
    580     lines.append('')
    581
    582     lines.append('')
    583
    584     lines.append('')
    585
    586     lines.append('')
    587
    588     lines.append('')
    589
    590     lines.append('')
    591
    592     lines.append('')
    593
    594     lines.append('')
    595
    596     lines.append('')
    597
    598     lines.append('')
    599
    600     lines.append('')
    601
    602     lines.append('')
    603
    604     lines.append('')
    605
    606     lines.append('')
    607
    608     lines.append('')
    609
    610     lines.append('')
    611
    612     lines.append('')
    613
    614     lines.append('')
    615
    616     lines.append('')
    617
    618     lines.append('')
    619
    620     lines.append('')
    621
    622     lines.append('')
    623
    624     lines.append('')
    625
    626     lines.append('')
    627
    628     lines.append('')
    629
    630     lines.append('')
    631
    632     lines.append('')
    633
    634     lines.append('')
    635
    636     lines.append('')
    637
    638     lines.append('')
    639
    640     lines.append('')
    641
    642     lines.append('')
    643
    644     lines.append('')
    645
    646     lines.append('')
    647
    648     lines.append('')
    649
    650     lines.append('')
    651
    652     lines.append('')
    653
    654     lines.append('')
    655
    656     lines.append('')
    657
    658     lines.append('')
    659
    660     lines.append('')
    661
    662     lines.append('')
    663
    664     lines.append('')
    665
    666     lines.append('')
    667
    668     lines.append('')
    669
    670     lines.append('')
    671
    672     lines.append('')
    673
    674     lines.append('')
    675
    676     lines.append('')
    677
    678     lines.append('')
    679
    680     lines.append('')
    681
    682     lines.append('')
    683
    684     lines.append('')
    685
    686     lines.append('')
    687
    688     lines.append('')
    689
    690     lines.append('')
    691
    692     lines.append('')
    693
    694     lines.append('')
    695
    696     lines.append('')
    697
    698     lines.append('')
    699
    700     lines.append('')
    701
    702     lines.append('')
    703
    704     lines.append('')
    705
    706     lines.append('')
    707
    708     lines.append('')
    709
    710     lines.append('')
    711
    712     lines.append('')
    713
    714     lines.append('')
    715
    716     lines.append('')
    717
    718     lines.append('')
    719
    720     lines.append('')
    721
    722     lines.append('')
    723
    724     lines.append('')
    725
    726     lines.append('')
    727
    728     lines.append('')
    729
    730     lines.append('')
    731
    732     lines.append('')
    733
    734     lines.append('')
    735
    736     lines.append('')
    737
    738     lines.append('')
    739
    740     lines.append('')
    741
    742     lines.append('')
    743
    744     lines.append('')
    745
    746     lines.append('')
    747
    748     lines.append('')
    749
    750     lines.append('')
    751
    752     lines.append('')
    753
    754     lines.append('')
    755
    756     lines.append('')
    757
    758     lines.append('')
    759
    760     lines.append('')
    761
    762     lines.append('')
    763
    764     lines.append('')
    765
    766     lines.append('')
    767
    768     lines.append('')
    769
    770     lines.append('')
    771
    772     lines.append('')
    773
    774     lines.append('')
    775
    776     lines.append('')
    777
    778     lines.append('')
    779
    780     lines.append('')
    781
    782     lines.append('')
    783
    784     lines.append('')
    785
    786     lines.append('')
    787
    788     lines.append('')
    789
    790     lines.append('')
    791
    792     lines.append('')
    793
    794     lines.append('')
    795
    796     lines.append('')
    797
    798     lines.append('')
    799
    800     lines.append('')
    801
    802     lines.append('')
    803
    804     lines.append('')
    805
    806     lines.append('')
    807
    808     lines.append('')
    809
    810     lines.append('')
    811
    812     lines.append('')
    813
    814     lines.append('')
    815
    816     lines.append('')
    817
    818     lines.append('')
    819
    820     lines.append('')
    821
    822     lines.append('')
    823
    824     lines.append('')
    825
    826     lines.append('')
    827
    828     lines.append('')
    829
    830     lines.append('')
    831
    832     lines.append('')
    833
    834     lines.append('')
    835
    836     lines.append('')
    837
    838     lines.append('')
    839
    840     lines.append('')
    841
    842     lines.append('')
    843
    844     lines.append('')
    845
    846     lines.append('')
    847
    848     lines.append('')
    849
    850     lines.append('')
    851
    852     lines.append('')
    853
    854     lines.append('')
    855
    856     lines.append('')
    857
    858     lines.append('')
    859
    860     lines.append('')
    861
    862     lines.append('')
    863
    864     lines.append('')
    865
    866     lines.append('')
    867
    868     lines.append('')
    869
    870     lines.append('')
    871
    872     lines.append('')
    873
    874     lines.append('')
    875
    876     lines.append('')
    877
    878     lines.append('')
    879
    880     lines.append('')
    881
    882     lines.append('')
    883
    884     lines.append('')
    885
    886     lines.append('')
    887
    888     lines.append('')
    889
    890     lines.append('')
    891
    892     lines.append('')
    893
    894     lines.append('')
    895
    896     lines.append('')
    897
    898     lines.append('')
    899
    900     lines.append('')
    901
    902     lines.append('')
    903
    904     lines.append('')
    905
    906     lines.append('')
    907
    908     lines.append('')
    909
    910     lines.append('')
    911
    912     lines.append('')
    913
    914     lines.append('')
    915
    916     lines.append('')
    917
    918     lines.append('')
    919
    920     lines.append('')
    921
    922     lines.append('')
    923
    924     lines.append('')
    925
    926     lines.append('')
    927
    928     lines.append('')
    929
    930     lines.append('')
    931
    932     lines.append('')
    933
    934     lines.append('')
    935
    936     lines.append('')
    937
    938     lines.append('')
    939
    940     lines.append('')
    941
    942     lines.append('')
    943
    944     lines.append('')
    945
    946     lines.append('')
    947
    948     lines.append('')
    949
    950     lines.append('')
    951
    952     lines.append('')
    953
    954     lines.append('')
    955
    956     lines.append('')
    957
    958     lines.append('')
    959
    960     lines.append('')
    961
    962     lines.append('')
    963
    964     lines.append('')
    965
    966     lines.append('')
    967
    968     lines.append('')
    969
    970     lines.append('')
    971
    972     lines.append('')
    973
    974     lines.append('')
    975
    976     lines.append('')
    977
    978     lines.append('')
    979
    980     lines.append('')
    981
    982     lines.append('')
    983
    984     lines.append('')
    985
    986     lines.append('')
    987
    988     lines.append('')
    989
    990     lines.append('')
    991
    992     lines.append('')
    993
    994     lines.append('')
    995
    996     lines.append('')
    997
    998     lines.append('')
    999
    1000    lines.append('')
  
```


Resultados

• Adaptación didáctica del material:

- $M \in \mathbb{R}^{(6+n) \times (6+n)}$ es la matriz de inercia generalizada, que describe cómo las masas y sus distribuciones afectan la aceleración \ddot{q} .
- $h \in \mathbb{R}^{6+n}$ representa las fuerzas no lineales, como:
 - **Fuerza centrífuga:** generada por el movimiento giratorio de las articulaciones.
 - **Fuerza de Coriolis:** describe cómo el movimiento de una parte del robot afecta dinámicamente a otras.
 - **Gravedad:** que actúa hacia el suelo.
- $S = \begin{bmatrix} 0_{n \times 6} \\ I_{6 \times 6} \end{bmatrix}$ es una matriz de selección que aplica los torques de articulaciones τ únicamente a las coordenadas actuadas. En este caso las primeras n filas representan las coordenadas del tronco, pero este no se ve afectado por ningún torque directo.
- $J(q)$ es el jacobiano de las coordenadas generalizadas, que proyecta las fuerzas externas f en las coordenadas del sistema.

2.1.3 Ejemplo Práctico

Imaginemos un robot cuadrúpedo con las siguientes características:

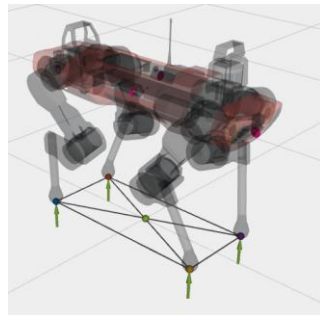
- Masa total del robot: $m = 50 \text{ kg}$.
- Aceleración gravitacional: $g = 9.81 \text{ m/s}^2$.
- El CoM se encuentra en $\mathbf{r}(q) = [0.0, 0.0, 0.5] \text{ m}$.
- Cada pata genera una fuerza de contacto f_i y se encuentran en:
 - Pata 1: $\mathbf{p}_1 = [-0.2, 0.1, 0.0] \text{ m}$
 - Pata 2: $\mathbf{p}_2 = [0.2, 0.1, 0.0] \text{ m}$
 - Pata 3: $\mathbf{p}_3 = [-0.2, -0.1, 0.0] \text{ m}$
 - Pata 4: $\mathbf{p}_4 = [0.2, -0.1, 0.0] \text{ m}$

Supongamos que las fuerzas normales son:

$$\mathbf{f}_1 = \mathbf{f}_2 = \mathbf{f}_3 = \mathbf{f}_4 = [0, 0, 122.625] \text{ N}$$

Esto compensa el peso del robot ($mg = 50 \times 9.81 = 490.5 \text{ N}$).

• Simulación



Objetivo final

- Generar un documento base de fácil entendimiento para los que quieran adentrarse en el mundo de los Legged Robot.
- Generar una pequeña base del funcionamiento de ROS con el fin de incentivar y facilitar su uso.

Referencias

1. Winkler, A. W. (2018). *Optimization-based motion planning for legged robots* (Tesis doctoral, ETH Zurich). Agile & Dexterous Robotics Lab, Institute for Robotics and Intelligent Systems. Recuperado de: <https://www.research-collection.ethz.ch/handle/20.500.11850/272432>

Gracias

Universidad Nacional de Colombia
