# FreeSandal

**樹莓派, 樹莓派之學習, 樹莓派之教育**

# STEM 隨筆：古典力學：運動學【五．四．B】

2018-05-04 | 懸鉤子 | 發表迴響

有人問：《**莎士比亞**》戲劇寫的好嗎？寫的很好，但是沒有什麼人讀！問：為什麼呢？？它用的是十六世紀的『古英文』。你是說『文外文』喔！！不是『**文外文**』，是那個時代的『**白話文**』。此時問者『**三條線**』：不是『過去的』都是『文□文』。答：要這麼講的話，那個五四之『**白話文**』也是『**過去的**』，現今**火紅**的是『唱給你聽』、『演給你看』以及『跳給你讚』，即使『後現代』、『火星語』都已經『**落伍**』，新潮流是『聽圖看話』...趕忙打斷來問：『**聽**』圖？『**看**』話？有沒有搞錯？？答： 不信！去 **LINE** 一下，**哩兜哉樣**！！

要說『秦始皇』推動『車同軌，書同文。』是好事嗎？再講到推動『國同語』呢？？試看『滿族』之『語言』和『文字』已經成為了『保護資產』，此事『康熙大帝』當時果真能得先見，料想知之的嘛 ！以此觀之

**漢賦、唐詩、宋詞、元曲、明清小說。**

，如果沒有人『**讀**』，自然就『**消逝**』時流之中，等待再次被『**人**』發現的吧！！

元代**王實甫**著《**西廂記**》—— **金聖嘆**評為『**第六才子書**』 —— 是『文言文』，這《西廂記》改自

唐．**元稹**．《**鶯鶯傳**》—— 《**會真記**》，更是『**古文言文**』，何妨當作『非現代』之『金星文』 Touch Touch

……

如此，自己吃過了豬肉，真憑實據，又哪會『**空口**』說『**白話**』的呢！就算有人『**踩盤子**』

摔杯子，真才實料，豈在意『**信口**』敢『**雌黃**』！！

——— 《字詞網絡： WORDNET 《五》 白話文 □○？○□！》

『思潮』運動也，蓋『精神歸屬』分合聚散現象也。

莫以為今日觸景撞期，故作此題外之言，實則人工智慧

# Artificial intelligence

**Artificial intelligence** (**AI**, also **machine intelligence**, **MI**) is intelligence demonstrated by machines, in contrast to the **natural intelligence** (**NI**) displayed by humans and other animals. In computer science AI research is defined as the study of "intelligent agents": any device that perceives its environment and takes actions that maximize its chance of successfully achieving its goals.[1] Colloquially, the term "artificial intelligence" is applied when a machine mimics "cognitive" functions that humans associate with other human minds, such as "learning" and "problem solving".[2]

The scope of AI is disputed: as machines become increasingly capable, tasks considered as requiring "intelligence" are often removed from the definition, a phenomenon known as the AI effect, leading to the quip, "AI is whatever hasn't been done yet."[3] For instance, optical character recognition is frequently excluded from "artificial intelligence", having become a routine technology.[4] Capabilities generally classified as AI as of 2017 include successfully understanding human speech,[5] competing at the highest level in strategic game systems (such as chess and Go[6]), autonomous cars, intelligent routing in content delivery network and military simulations.

Artificial intelligence was founded as an academic discipline in 1956, and in the years since has experienced several waves of optimism,[7][8] followed by disappointment and the loss of funding (known as an "AI winter"),[9][10] followed by new approaches, success and renewed funding.[8][11] For most of its history, AI research has been divided into subfields that often fail to communicate with each other.[12] These sub-fields are based on technical considerations, such as particular goals (e.g. "robotics" or "machine learning"),[13] the use of particular tools ("logic" or "neural networks"), or deep philosophical differences.[14][15][16] Subfields have also

been based on social factors (particular institutions or the work of particular researchers).[12]

The traditional problems (or goals) of AI research include reasoning, knowledge representation, planning, learning, natural language processing, perception and the ability to move and manipulate objects.[13] General intelligence is among the field's long-term goals.[17] Approaches include statistical methods, computational intelligence, and traditional symbolic AI. Many tools are used in AI, including versions of search and mathematical optimization, neural networksand methods based on statistics, probability and economics. The AI field draws upon computer science, mathematics, psychology, linguistics, philosophy and many others.

The field was founded on the claim that human intelligence "can be so precisely described that a machine can be made to simulate it".[18] This raises philosophical arguments about the nature of the mind and the ethics of creating artificial beings endowed with human-like intelligence which are issues that have been explored by myth, fiction and philosophy since antiquity.[19] Some people also consider AI to be a danger to humanity if it progresses unabatedly.[20]Others believe that AI, unlike previous technological revolutions, will create a risk of mass unemployment.[21]

In the twenty-first century, AI techniques have experienced a resurgence following concurrent advances in computer power, large amounts of data, and theoretical understanding; and AI techniques have become an essential part of thetechnology industry, helping to solve many challenging problems in computer science.[22][11]

大浪不遠。然而善用數理『軟體工具』學習者寥寥可數，遑論寫程式也。難到想

機器都有智慧。

時，再談教育『革命』乎？

屆時人文恐果已被排除在外呦！其真能與德里達論辯矣！？

當被問及「什麼是解構主義？」德里達回答說，「我這個問題沒有簡單的回答。我所有的文章都試圖把它帶出比這更可怕的問題」。德里達認為，解構必然是複雜的，難以解釋，因為

它積極地批評了需要用語言來解釋一切。

或將及香山之『非非半明』的境界哩？！

《花非花》白居易

花非花，霧非霧，
夜半來，天明去，
來如春夢無多時，
去似朝雲無覓處。

香山居士這首詩別出心裁，令人想入非非。莫非

美人花，花非花，似花花解語。
彩雲霧，霧非霧，比霧霧生霞。
春夢恐醒，韶光將逝，如電亦如霧。
朝雲易散，彩霞難留，來去無覓處。

。引人『  』悟吾心乎？？感嘆人生幾何耶！！

幾人曾賞霧中花？
飄香方知花是花！
春夢朝雲花中霧！
何時才曉霧是霧？

暑中偶偶風雨突至，雷電交加，正是讚嘆自然現象之際，恰合說此萬花尺之時：

萬花筒非筒？萬花尺非尺！
那位乘天光！千度百回！！

這位伴銀河！百媚千嬌！！
莫要問：乾坤尺筒何時有？？
無須答：天地筒尺幾曾無！！

——— 摘自《光的世界：派生科學計算五》

還不快一步一腳印：

# system

# Introduction

The System class manages the simulation (integration) of a system whose equations are given by KanesMethod.

Many of the attributes are also properties, and can be directly modified.

Here is the procedure for using this class.

1. *specify your options either via the constructor or via the attributes.*
2. *optionally, call* `generate_ode_function()` *if you want to customize how the ODE function is generated.*
3. *call* `integrate()` *to simulate your system.*

## Examples

The simplest usage of this class is as follows. First, we need a KanesMethod object on which we have already invoked `kanes_equations()`:

```
km = KanesMethod(...)
km.kanes_equations(force_list, body_list)
times = np.linspace(0, 5, 100)
sys = System(km, times=times)
sys.integrate()
```

In this case, we use defaults for the numerical values of the constants, specified quantities, initial conditions, etc. You probably won't like these defaults. You can also specify such values

via constructor keyword arguments or via the attributes:

```
sys = System(km,
             initial_conditions={dynamicsymbol('q1'): 0.5},
             times=times)
sys.constants = {symbol('m'): 5.0}
sys.integrate()
```

To double-check the constants, specifieds, states and times in your problem, look at these properties:

```
sys.constants_symbols
sys.specifieds_symbols
sys.states
sys.times
```

In this case, the System generates the numerical ode function for you behind the scenes. If you want to customize how this function is generated, you must call `generate_ode_function` on your own:

```
sys = System(KM)
sys.generate_ode_function(generator='cython')
sys.integrate()
```

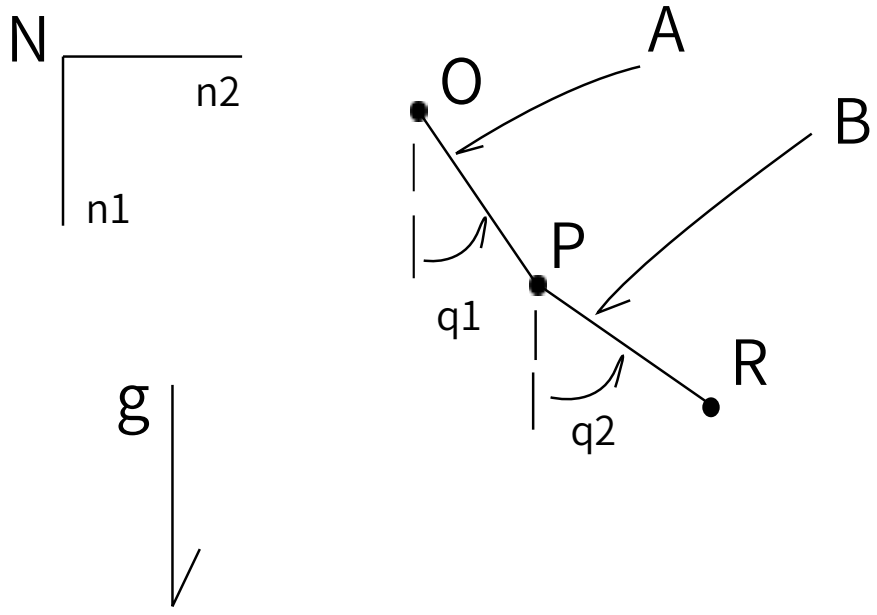
清風浮雲伴得晚霞歸◎

# A Double Pendulum Example

## Problem Description

The double pendulum is a classic example of a chaotic system. It is a well documented and understood example. Read [here](wp>Double_pendulum) for more details.

This example has 2 particles of the same mass, on massless links of the same length.




Double Pendulum Example

※ 補耶

```
from __future__ import print_function, division
from sympy import init_printing
init_printing(use_latex='mathjax', pretty_print=False)
```

```
from sympy import*
from sympy.physics.mechanics import *
```

```
q1, q2 = dynamicsymbols('q1 q2')
q1d, q2d = dynamicsymbols('q1 q2', 1)
u1, u2 = dynamicsymbols('u1 u2')
u1d, u2d = dynamicsymbols('u1 u2', 1)
l, m, g = symbols('l m g')

N = ReferenceFrame('N')
A = N.orientnew('A', 'Axis', [q1, N.z])
B = N.orientnew('B', 'Axis', [q2, N.z])

A.set_ang_vel(N, u1 * N.z)
B.set_ang_vel(N, u2 * N.z)

O = Point('O')
P = O.locatenew('P', l * A.x)
R = P.locatenew('R', l * B.x)

O.set_vel(N, 0)
P.v2pt_theory(O, N, A)
R.v2pt_theory(P, N, B)

ParP = Particle('ParP', P, m)
ParR = Particle('ParR', R, m)

kd = [q1d - u1, q2d - u2]
FL = [(P, m * g * N.x), (R, m * g * N.x)]
BL = [ParP, ParR]

KM = KanesMethod(N, q_ind=[q1, q2], u_ind=[u1, u2], kd_eqs=kd)
```

```
(fr, frstar) = KM.kanes_equations(FL, BL)
kdd = KM.kindiffdict()
mm = KM.mass_matrix_full
fo = KM.forcing_full
qudots = mm.inv() * fo
qudots = qudots.subs(kdd)
qudots.simplify()
qudots
```

$$\left[\begin{matrix} u_1(t) \\ u_2(t) \\ \frac{1}{2l\left(\cos^2\left(q_1(t) - q_2(t)\right) - 2\right)}\left(g\sin\left(q_1(t) - 2q_2(t)\right) + 3g\sin\left(q_1(t)\right) + l{u_1}^2(t)\sin\left(2q_1(t) - 2q_2(t)\right) + 2l{u_2}^2(t)\sin\left(q_1(t) - q_2(t)\right)\right) \\ \frac{1}{l\left(\cos^2\left(q_1(t) - q_2(t)\right) - 2\right)}\left(-g\sin\left(2q_1(t) - q_2(t)\right) + g\sin\left(q_2(t)\right) - 2l{u_1}^2(t)\sin\left(q_1(t) - q_2(t)\right) - \frac{l}{2}{u_2}^2(t)\sin\left(2q_1(t) - 2q_2(t)\right)\right) \end{matrix}\right]$$

```
from numpy import array, linspace
from pydy.system import System

sys = System(KM,
             constants={m:1.0, g:9.81, l:1.0},
             initial_conditions={q1: 0.1, q2:0.2, u1:0.0, u2: 0.0},
             times=linspace(0.0, 10.0, 1000))
```

```
y = sys.integrate()
```

```
%matplotlib inline
import matplotlib.pyplot as plt
plt.plot(sys.times, y)
plt.legend((str(q1), str(q2), str(u1), str(u2)))
plt.show()
```