Computational Physics

Topic 03 — Computational Problems involving Marko Chains

Lecture 02 — The Collector Problem

Dr Kieran Murphy

Department of Computing and Mathematics,
Waterford IT.
(kmurphy@wit.ie)

Autumn Semester, 2021

RESOURCE OUTLINE LABEL

- Problem statement
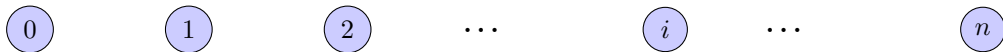- Sample run

Markov chain approach

> Model

- **State**: i, $i = 0, \ldots, n$, where i is number of collected coupons.

  Have i of the n available coupons, so $(n - i)$ coupons are unseen.

- Initial state is 0. State n is terminal.
- **Stage**: Number of coupons purchased.

  How long does it take to travel from 0 to n?

0 1 2 $\cdots$ i $\cdots$ n
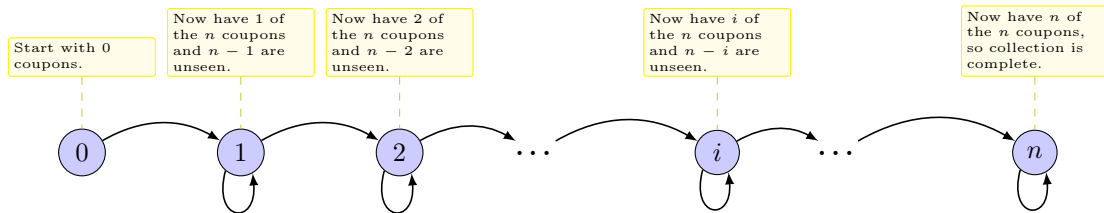
Markov chain approach

- **State**: i, $i = 0, \ldots, n$, where i is number of collected coupons.
  Have i of the n available coupons, so $(n - i)$ coupons are unseen.
- Initial state is 0. State n is terminal.
- **Stage**: Number of coupons purchased.
  How long does it take to travel from 0 to n?

Start with 0 coupons.

Now have 1 of the n coupons and $n - 1$ are unseen.

Now have 2 of the n coupons and $n - 2$ are unseen.

Now have i of the n coupons and $n - i$ are unseen.

Now have n of the n coupons, so collection is complete.

0 1 2 $\cdots$ i $\cdots$ n

Markov chain approach

> Model

- **State**: i, $i = 0, \ldots, n$, where i is number of collected coupons.
  Have i of the n available coupons, so $(n - i)$ coupons are unseen.
- Initial state is 0. State n is terminal.
- **Stage**: Number of coupons purchased.
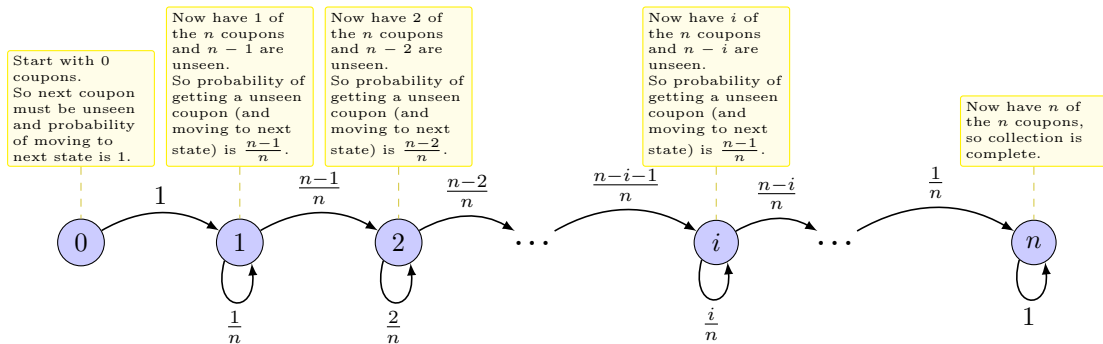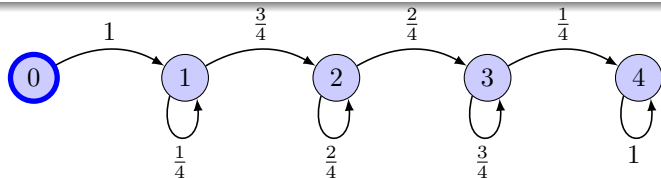 *How long does it take to travel from 0 to n?*
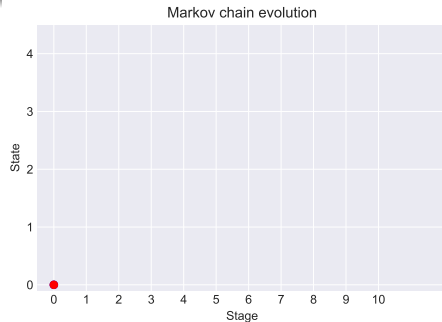
Markov chain approach

> Model

- **State**: i, $i = 0, \ldots, n$, where i is number of collected coupons.

  Have i of the n available coupons, so $(n - i)$ coupons are unseen.

- Initial state is 0. State n is terminal.

- **Stage**: Number of coupons purchased.

  How long does it take to travel from 0 to n?

Start with 0 coupons. So next coupon must be unseen and probability of moving to next state is 1.

Now have 1 of the n coupons and $n - 1$ are unseen. So probability of getting a unseen coupon (and moving to next state) is $\frac{n-1}{n}$.

Now have 2 of the n coupons and $n - 2$ are unseen. So probability of getting a unseen coupon (and moving to next state) is $\frac{n-2}{n}$.

Now have i of the n coupons and $n - i$ are unseen. So probability of getting a unseen coupon (and moving to next state) is $\frac{n-1}{n}$.

Now have n of the n coupons, so collection is complete.

Viewing our first simulation run as a Markov chain . . .

count: 0   found: collected: set()

Markov chain evolution

Viewing our first simulation run as a Markov chain . . .
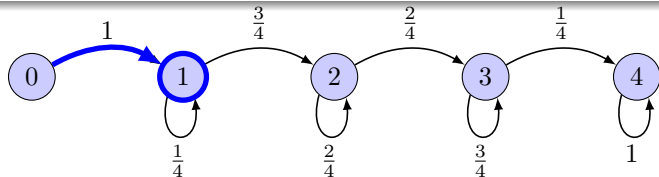

```
count:    0       found:         collected: set()
count:    1       found: {2}     collected: {2}
```
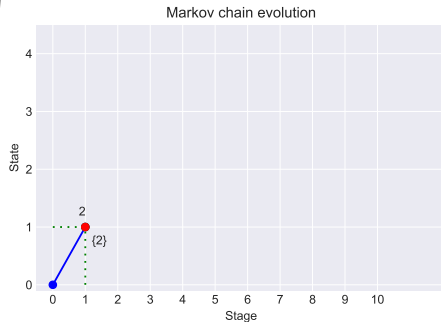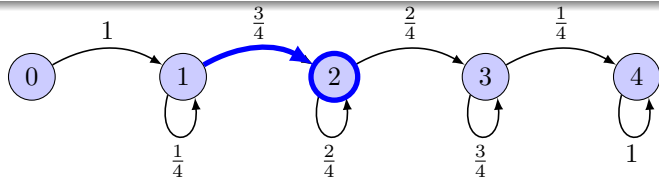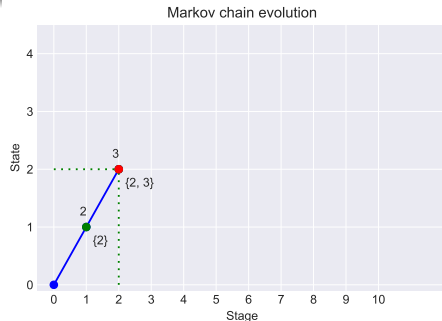
Viewing our first simulation run as a Markov chain . . .
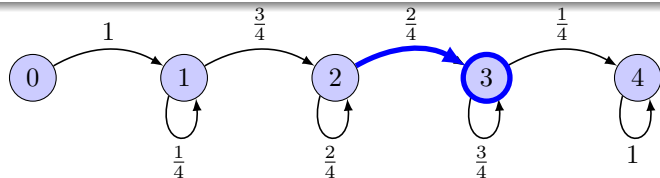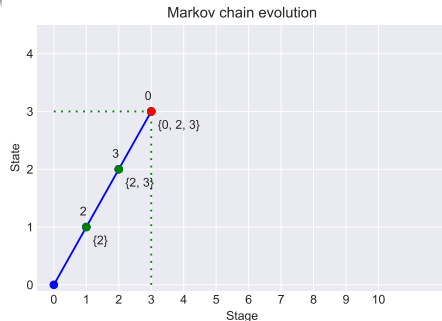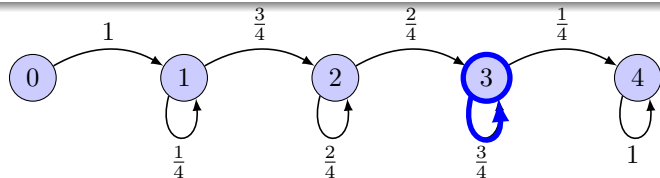


```
count:    0        found:        collected: set()
count:    1        found: {2}    collected: {2}
count:    2        found: {3}    collected: {2, 3}
```

Viewing our first simulation run as a Markov chain ...

Markov chain evolution

```
count:    0       found:         collected: set()
count:    1       found: {2}     collected: {2}
count:    2       found: {3}     collected: {2, 3}
count:    3       found: {0}     collected: {0, 2, 3}
```
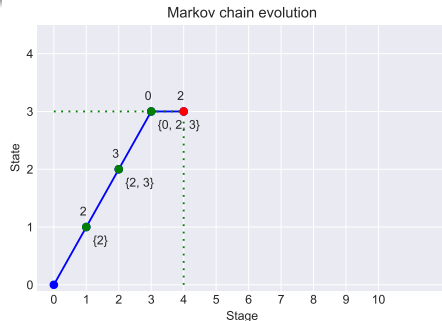
Viewing our first simulation run as a Markov chain . . .


```
count:      0       found:        collected: set()
count:      1       found: {2}    collected: {2}
count:      2       found: {3}    collected: {2, 3}
count:      3       found: {0}    collected: {0, 2, 3}
count:      4       found: {2}    collected: {0, 2, 3}
```
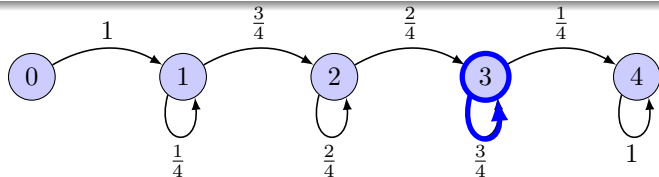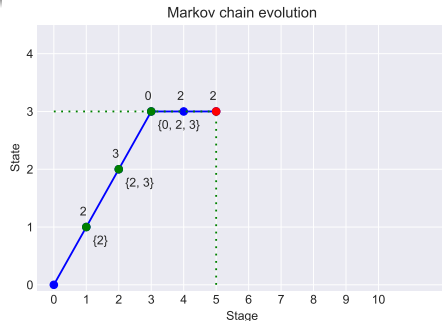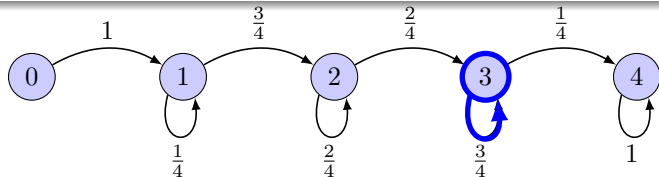
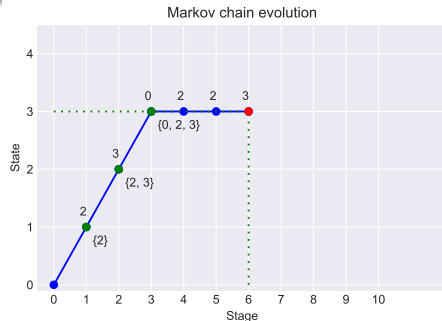# Viewing our first simulation run as a Markov chain . . .

count:	0	found:	collected: set()
count:	1	found: {2}	collected: {2}
count:	2	found: {3}	collected: {2, 3}
count:	3	found: {0}	collected: {0, 2, 3}
count:	4	found: {2}	collected: {0, 2, 3}
count:	5	found: {2}	collected: {0, 2, 3}

Viewing our first simulation run as a Markov chain . . .


```
count :     0      found :         collected : set ()
count :     1      found : {2}     collected : {2}
count :     2      found : {3}     collected : {2, 3}
count :     3      found : {0}     collected : {0, 2, 3}
count :     4      found : {2}     collected : {0, 2, 3}
count :     5      found : {2}     collected : {0, 2, 3}
count :     6      found : {3}     collected : {0, 2, 3}
```

Viewing our first simulation run as a Markov chain . . .


```
count:    0    found:          collected: set()
count:    1    found: {2}      collected: {2}
count:    2    found: {3}      collected: {2, 3}
count:    3    found: {0}      collected: {0, 2, 3}
count:    4    found: {2}      collected: {0, 2, 3}
count:    5    found: {2}      collected: {0, 2, 3}
count:    6    found: {3}      collected: {0, 2, 3}
count:    7    found: {0}      collected: {0, 2, 3}
```
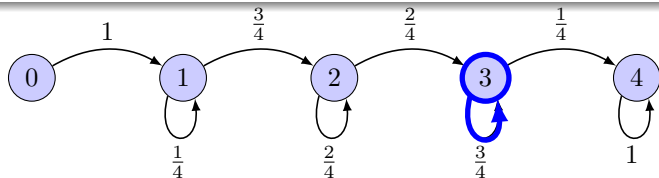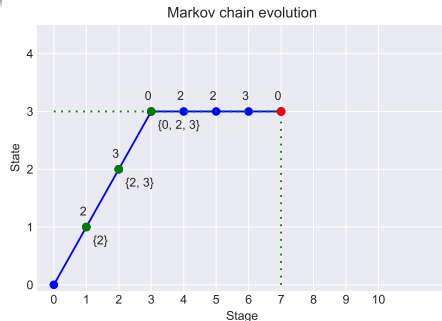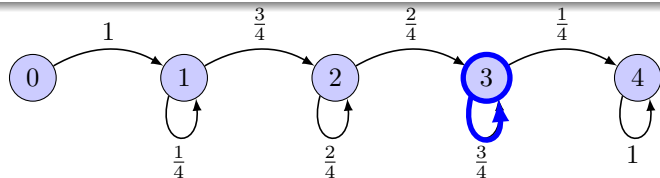
Viewing our first simulation run as a Markov chain . . .
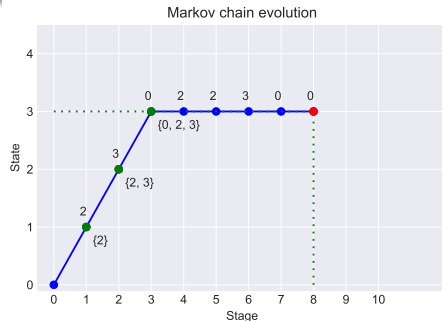


```
count:     0     found:       collected: set()
count:     1     found: {2}   collected: {2}
count:     2     found: {3}   collected: {2, 3}
count:     3     found: {0}   collected: {0, 2, 3}
count:     4     found: {2}   collected: {0, 2, 3}
count:     5     found: {2}   collected: {0, 2, 3}
count:     6     found: {3}   collected: {0, 2, 3}
count:     7     found: {0}   collected: {0, 2, 3}
count:     8     found: {0}   collected: {0, 2, 3}
```
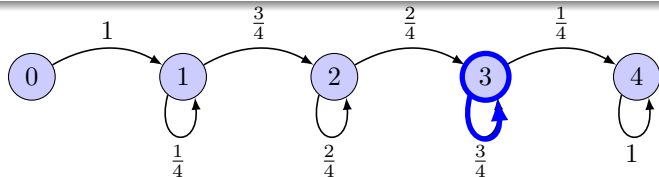
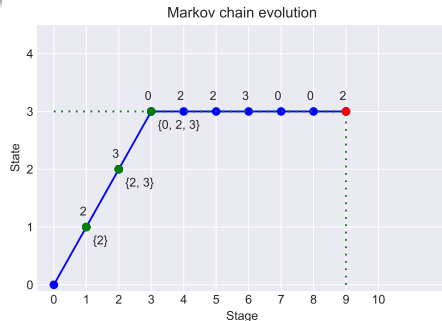# Viewing our first simulation run as a Markov chain . . .

Markov chain evolution

```
count:    0      found:         collected: set()
count:    1      found: {2}     collected: {2}
count:    2      found: {3}     collected: {2, 3}
count:    3      found: {0}     collected: {0, 2, 3}
count:    4      found: {2}     collected: {0, 2, 3}
count:    5      found: {2}     collected: {0, 2, 3}
count:    6      found: {3}     collected: {0, 2, 3}
count:    7      found: {0}     collected: {0, 2, 3}
count:    8      found: {0}     collected: {0, 2, 3}
count:    9      found: {2}     collected: {0, 2, 3}
```
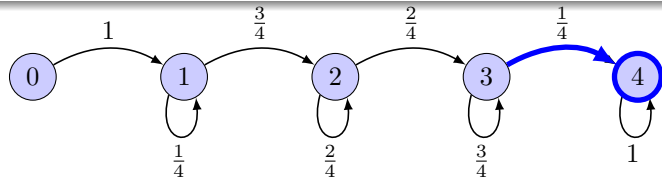
Viewing our first simulation run as a Markov chain ...


```
count:     0     found:          collected: set()
count:     1     found: {2}      collected: {2}
count:     2     found: {3}      collected: {2, 3}
count:     3     found: {0}      collected: {0, 2, 3}
count:     4     found: {2}      collected: {0, 2, 3}
count:     5     found: {2}      collected: {0, 2, 3}
count:     6     found: {3}      collected: {0, 2, 3}
count:     7     found: {0}      collected: {0, 2, 3}
count:     8     found: {0}      collected: {0, 2, 3}
count:     9     found: {2}      collected: {0, 2, 3}
count:    10     found: {1}      collected: {0, 1, 2, 3}
```
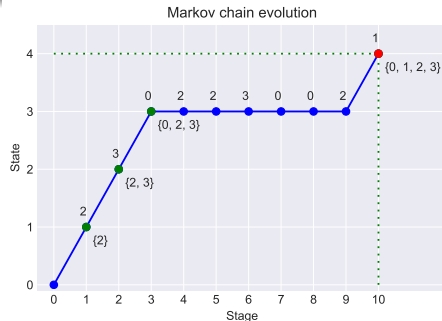
Number of stages needed = 10