

# 2015 年度 大問 4

hari64boli64 (hari64boli64@gmail.com)

2023 年 5 月 8 日

## 1 問題

$$I = [0, 1], I_1 = [0, \frac{1}{2}], I_2 = [\frac{1}{2}, 1]$$

$$f(x) = \begin{cases} x + \frac{1}{2} & (x \in I_1) \\ 2 - 2x & (x \in I_2) \end{cases}$$

## 2 解答

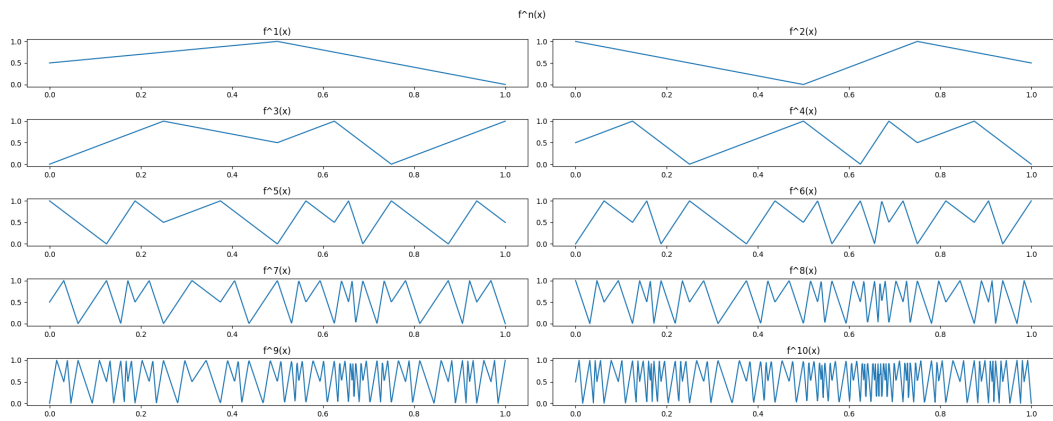


図 1  $f^n(x)$

### (1)

$J$  は閉区間であって、閉集合ではないことに注意。  
この点さえ分かれば、中間値の定理より自明。

(2)

(2-1)

不明

(2-2)

自明

(2-3)

存在性は不明。

一意性は (2-2) より自明。

(3)

隣接行列の意味より自明。(大嘘)

(4)

対角化して求めればよい。(todo)

実は、フィボナッチ数列が出てくる。(おまけ参照)

よって、ビネの公式より、求める値は  $\frac{1+\sqrt{5}}{2}$  となる。

### 3 知識

ビネの公式は、以下の通り。[\[1\]](#)

$$F_n = \frac{1}{\sqrt{5}} \left\{ \left( \frac{1+\sqrt{5}}{2} \right)^n - \left( \frac{1-\sqrt{5}}{2} \right)^n \right\}$$

### 4 おまけ

#### ソースコード 1 calculateLimit

```
1 import math
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5
6 def _f(x):
7     assert 0 <= x <= 1
```

```

8     if x <= 1 / 2:
9         return x + 1 / 2
10    else:
11        return 2 - 2 * x
12
13
14    def f(n, x):
15        for _ in range(n):
16            x = _f(x)
17        return x
18
19
20    def visFn():
21        fig, axes = plt.subplots(5, 2, figsize=(20, 8))
22        for n in range(1, 10 + 1):
23            Xs = np.linspace(0, 1, 2**10)
24            axes[(n - 1) // 2][(n - 1) % 2].plot(Xs, [f(n, x) for x
25                in Xs])
26            axes[(n - 1) // 2][(n - 1) % 2].set_title(f"f^{n}(x)")
27        fig.suptitle("f^n(x)")
28        plt.tight_layout()
29        plt.savefig("4.png")
30
31    def calcLogNp():
32        A = np.array([[0, 1], [1, 1]], dtype=object)
33        for n in range(1, 10):
34            An = np.linalg.matrix_power(A, n)
35            print(f"n: {n}, A^n: {An.tolist()}")
36            print(f"trace: {np.trace(An)}")
37        print("=" * 10)
38        print(
39            f"limit of logN(p)/p:",
40            math.log(np.trace(np.linalg.matrix_power(A, 1 << 20))) /
41                (1 << 20),
42        )
43        print(f"{math.log((1+math.sqrt(5))/2)=}")
44
45    def main():
46        visFn()
47        calcLogNp()
48
49
50    if __name__ == "__main__":
51        main()

```

## ソースコード 2 result

```
1  n: 1, A^n: [[0, 1], [1, 1]]
2  trace: 1
3  n: 2, A^n: [[1, 1], [1, 2]]
4  trace: 3
5  n: 3, A^n: [[1, 2], [2, 3]]
6  trace: 4
7  n: 4, A^n: [[2, 3], [3, 5]]
8  trace: 7
9  n: 5, A^n: [[3, 5], [5, 8]]
10 trace: 11
11 n: 6, A^n: [[5, 8], [8, 13]]
12 trace: 18
13 n: 7, A^n: [[8, 13], [13, 21]]
14 trace: 29
15 n: 8, A^n: [[13, 21], [21, 34]]
16 trace: 47
17 n: 9, A^n: [[21, 34], [34, 55]]
18 trace: 76
19 =====
20 limit of logN(p)/p: 0.4812118250596034
21 math.log((1+math.sqrt(5))/2)=0.48121182505960347
```

## 参考文献

- [1] 高校数学の美しい物語.”フィボナッチ数列の7つの性質（一般項・黄金比・互いに素）”.2021年8月17日.<https://manabitimes.jp/math/643>