

平成29年度 知能機械情報学専攻

大学院入学試験問題

「知能機械情報学（科目）」

試験日時：平成28年8月22日（月）14：00～16：00

注意事項

1. 試験開始の合図があるまで、この問題冊子を開かないこと。
2. 問題は3題出題されている。問題1（必答問題）は必ず解答し、問題2Aおよび問題2B（選択問題）から1題を選択して解答すること。
3. 問題の解釈に複数の可能性が考えられる場合は、適宜言葉の定義や条件などを付加して解答してよい。
4. 問題冊子に落丁、乱丁、あるいは印刷不鮮明な箇所があれば申し出ること。
5. 答案用紙は2枚配布される。枚数を確認し、過不足があれば申し出ること。問題ごとに1枚の答案用紙を用いて解答すること。解答を表面で書ききれない場合は裏面を使用しても構わない。その際は裏面にも解答した旨を表面に記入すること。
6. 答案用紙の指定された箇所に、科目名の「知能機械情報学（科目）」、修士・博士の別、受験番号、その答案用紙で解答する問題番号を記入すること。これらが記入漏れの場合は採点されないことがある。
7. 解答に関係のない記号や符号を記入した答案は無効となる。
8. 答案用紙は、解答ができなかった問題についても、科目名、修士・博士の別、受験番号、問題番号を記入し、2枚全部を提出すること。
9. 下書きは問題冊子の草稿用のページを用いること。
10. この問題冊子にも受験番号を記入し提出すること。

受験番号	
------	--

上欄に受験番号を記入すること。

草稿用紙
(切り取らないこと)

草稿用紙
(切り取らないこと)

問題 1 (必答問題)

問 1. 図 1 のような倒立振り子を考える. なめらかで水平な床上に質量 M_1 の台車が置かれている. 台車は外力 f を受けて x 軸方向に移動する. 一様な細い棒の片端が台車と回転関節で連結され, xy 平面内で回転する. 棒の質量を M_2 , 長さを L , 回転関節の位置を $(x, 0)$, 鉛直方向からの傾きを θ とする. 摩擦は無視できるものとする. 重力加速度を g とする. 以下の間に答えよ.

- (1) 棒の重心まわりの慣性モーメント I を求めよ.
- (2) ラグランジアンを用いて倒立振り子の運動方程式を求めよ.

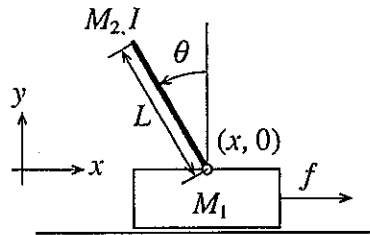


図 1

問 2. 論理回路に関する以下の間に答えよ. ただし, $x_1, x_2, y \in \{0, 1\}$.

- (1) 図 2 のような, 入力 x_1, x_2 , 出力 y の NAND ゲートの真理値表を書け.

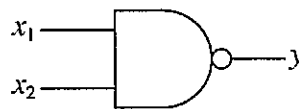


図 2

- (2) 図 2 の NAND ゲートのみで 2 入力の OR ゲートが構成できることを論理式で示せ. また, その回路図を描け.
- (3) 図 3 のような, 固定バイアス入力 $x_0 = 1$, 入力 x_1, x_2 の人工ニューロンの出力 y が, 以下の式で表現される. この人工ニューロンが NAND ゲートとして機能するための重み $w_i (i = 0, 1, 2)$ を例示せよ.

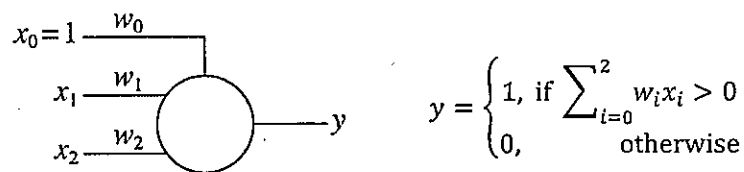


図 3

問 3. 以下の語句について簡潔に説明せよ.

- (1) オペレーティングシステムの「プロセス」と「スレッド」
- (2) 画像処理における「ガウシアンフィルタ」
- (3) デジタル信号処理における「サンプリング定理」

草稿用紙
(切り取らないこと)

問題 2 A (選択問題)

図 1 に示す巻上機 (winch), ロープ (rope), かご (basket) からなる昇降機を考える. 巻上機の半径を r , 慣性モーメントを J , 駆動トルクを τ , かごの質量を m_1 とする. 以下の間に答えよ. なお, 簡単のため, ロープの質量は考えなくてよい. また, 摩擦も無視してよい.

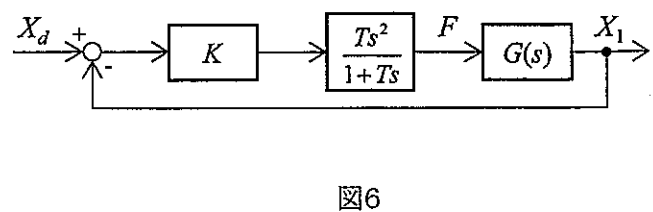
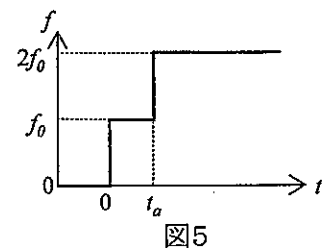
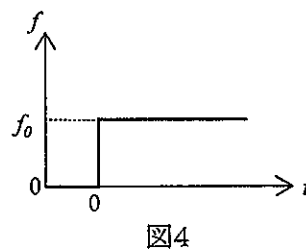
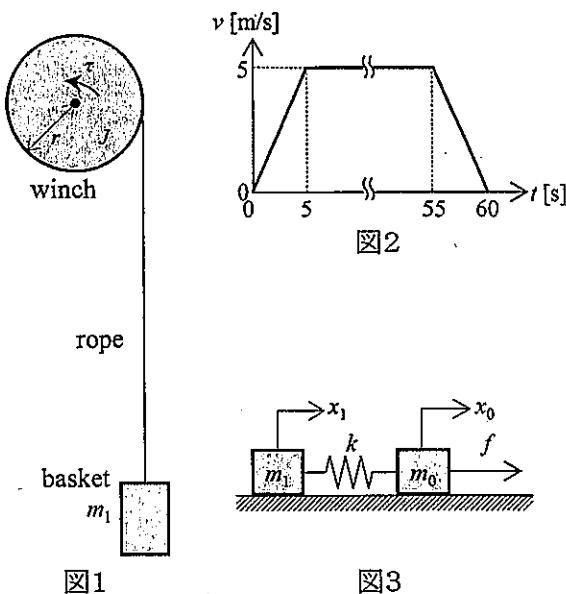
問 1. 図 2 のように, かごの速度 v [m/s] を制御する. $J=100$ [kg m²], $r=0.80$ [m], $m_1=1000$ [kg] とするとき, 巻上機の毎分の最大回転数 [rpm] と τ の最大値 [N m] を有効数字 2 桁で求めよ. なお, 重力加速度は 9.8 [m/s²] とする. ロープの伸縮は考えなくてよい.

問 2. 高層ビルの昇降機の運動では, ロープの弾性の影響を無視できなくなる. この影響をモデル化するため, 図 3 のように, 水平面上において, 質量 m_0, m_1 の質点とばね定数 k のばねからなる系を考え, m_0 に力 f を加える. x_0 と x_1 は, それぞれ, m_0 と m_1 のつりあいの静止位置からの変位とする.

- (1) 図 3 の 2 質点系の運動方程式を記せ.
- (2) 上記 (1) の運動方程式において, ロープの伸びを $\Delta x (=x_0 - x_1)$ とすると, $\Delta \ddot{x} + \omega_0^2 \Delta x = f/m_0$ と表せる. 式中の ω_0 を求めよ. なお, ω_0 は以下の設問 (問 3 含む) で用いてよい.
- (3) 図 4 に示すように, 時刻 $t=0$ で $f=f_0$ のステップ状の力を加えた. このとき, Δx の時間変化を求め, 図示せよ.
- (4) 図 5 に示すように, 時刻 $t=0$ で $f=f_0$ のステップ状の力を加えた後, さらに $t=t_a$ で $f=2f_0$ のステップ状の力を加えることで, Δx の振動成分を消失させたい. このとき, t_a を求めよ.

問 3. 図 3 のモデルで m_1 の位置制御を考える.

- (1) 問 2 の (1) において, 入力を f , 出力を x_1 とするとき, 伝達関数 $G(s) = X_1(s)/F(s)$ を求めよ. なお, $F(s)$ と $X_1(s)$ は, それぞれ, $f(t)$ と $x_1(t)$ のラプラス変換である.
- (2) 前問の $G(s)$ に対し, 図 6 に示すように, フィードバック制御系を構成した. ただし, X_d は目標値 $x_d(t)$ のラプラス変換, K と T は定数である. この系が安定となるための条件を記せ.



草稿用紙
(切り取らないこと)

問題 2B (選択問題)

問 1. C 言語プログラミングに関して以下の問に答えよ。

- (1) 自然数 1 から n までの総和 S_n は以下のように定義される。 S_n を値として返す関数 `int S(int n)` を再帰関数を用いる場合と用いない場合の 2 通り書け。

$$S_n = S_{n-1} + n, \quad \text{ただし } S_1 = 1$$

- (2) (1) の再帰を用いた関数 `int S(int n)` は n が増えると問題が起こる場合がある。どのような問題が起こるか、また、なぜ起こるか述べよ。
- (3) 以下のプログラムがある。 `apply(add, x, y)` は x と y の和を返し、 `apply(mul, x, y)` は x と y の積を返す。以下の下線部を埋めよ。

```
typedef int (*func)(int, int);
int add(int x, int y) { return x + y; }
int mul(int x, int y) { return x * y; }
int apply(func f, int x, int y) { return _____ ; }
```

問 2. リスト 1 は学習を行うプログラムを Python 言語で記述したものである。

関数 `update()` は重み配列 w , 入力データ配列 x , 学習率 u , 正解ラベル y に対して以下の式で w を更新する。

$$w = w + uyx \tag{1}$$

また、関数 `predict()` は関数 $g(a) = \begin{cases} +1 & (a \geq 0) \\ -1 & (a < 0) \end{cases}$ を用いて

$$\text{predict}(w, x) = g\left(\sum_{i=0}^{d-1} w_i x_i\right) \tag{2}$$

を計算する。ただし d は配列 w と x の長さとする。

- (1) リスト 1 中の (A) を埋めよ。
- (2) このプログラムを実行した際の出力を記せ。
- (3) リスト 1 中の (B) を埋め、式 (2) の \sum 記号に対応する関数 `sum()` を定義せよ。
- (4) リスト 1 中の (C) は式 (2) の $\sum_{i=0}^{d-1} w_i x_i$ に対応する。上で定義した関数 `sum()` を用いたできるだけ短いプログラムで (C) を埋めよ。新たに関数や変数を定義し利用しても良い。

問 3. リスト 1 で $xs = [[0,0,1], [0,1,1], [1,0,1], [1,1,1]]$, $ys = [-1,1,1,-1]$ とする。

- (1) このプログラムを実行すると何が起こるか、またその理由を述べよ。
- (2) この xs , ys についても学習が成功するようにプログラムを改良したい。そのためのなるべく簡潔なニューラルネットワークを図示せよ。また、その重み配列の更新方法を 5 行程度で説明せよ。


```

#!/usr/bin/env python

u = 0.25
def main():
    w = [0.125, 0.125, 0.25]
    o = [0, 0, 0, 0]
    xs = [[0, 0, 1], [0, 1, 1], [1, 0, 1], [1, 1, 1]]
    ys = [-1, -1, -1, 1]
    while o != ys:
        for j in [0, 1, 2, 3]:
            o[j] = predict(w, xs[j])
            if o[j] != ys[j]:
                w = update(w, xs[j], ys[j])
        print(o)
    print(w)

def update(w, x, y):
    (A)
    -----

def sum (f, a, b):
    if a > b:
        return 0
    else:
        return (f(a) + (B) )
    -----

def predict(w, x):
    (C)
    -----

    if a >= 0:
        return 1
    else:
        return -1

main()

```

リスト1

草稿用紙
(切り取らないこと)

Mechano-Informatics (Subject)

Date : 2016, August 22 (Mon), 14:00 – 16:00

Instructions:

- 0) Answers should be written either in Japanese or English.
- 1) Do not open this problem booklet until the start of the examination is announced.
- 2) Three problems are provided. Solve Problem 1 (Compulsory), and solve either Problem 2A or Problem 2B (Required Elective).
- 3) When you have multiple interpretations of a problem statement, you may clarify your interpretation by introducing adequate definitions and/or conditions in your answer.
- 4) If you find missing, misplaced, and/or unclearly printed pages in the problem booklet, notify the examiner.
- 5) Two answer sheets are provided. Check the number of them, and if you find excess or deficiency, notify the examiner. You must use a separate sheet for each problem. When you run short of space for your answer on the front side of the answer sheet, you may use the back side by clearly stating so in the front side.
- 6) In the designated blanks at the top of each answer sheet, write examination name “Mechano-Informatics (Subject)”, “Master” or “Doctor”, your applicant number, and the problem number. Failure to fill up these blanks may void your test score.
- 7) An answer sheet is regarded as invalid if you write marks and/or symbols unrelated to the answer.
- 8) Even if the answer sheet(s) is blank, submit all answer sheets with examination name, “Master” or “Doctor”, your applicant number, and the problem number.
- 9) Use the blank pages in the problem booklet for your draft.
- 10) Fill in the blank below with your applicant number, and submit this booklet.

Applicant number:

MEMO

(Do not detach this page)

MEMO

(Do not detach this page)

Problem 1 (Compulsory)

P. 1. Figure 1 shows an inverted pendulum. A cart with mass M_1 is on a smooth level surface. An external force f moves the cart along the x axis. One end of a uniform thin rod is connected to the cart with a pivot joint and rotates in the xy -plane. The mass of the rod is M_2 , the length is L , the position of the pivot joint is $(x, 0)$, and the angle from the vertical direction is θ . Friction is negligible. The gravitational acceleration is g . Answer the following questions.

- (1) Obtain I , the moment of inertia of the rod about the rod's center of mass.
- (2) Derive equation(s) of motion of the inverted pendulum by using the Lagrangian.

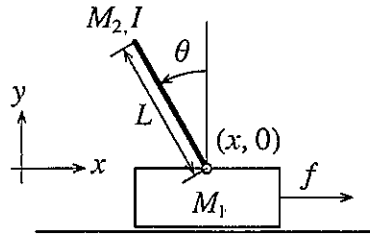


Fig. 1

P. 2. Answer the following questions about logic circuits, where $x_1, x_2, y \in \{0,1\}$.

- (1) Write a truth table of a NAND gate with inputs x_1, x_2 and an output y shown in Fig. 2.

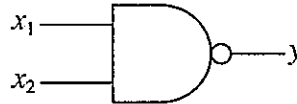


Fig. 2

- (2) Show that a two-input OR gate can be formed only using NAND gates shown in Fig. 2 with a logical formula. Then draw its logic circuit diagram.
- (3) An artificial neuron with a fixed bias input $x_0 = 1$, and inputs x_1, x_2 shown in Fig. 3 has an output y subject to the following equation. Provide an example of the weight values $w_i (i = 0,1,2)$ so that the artificial neuron works as a NAND gate.

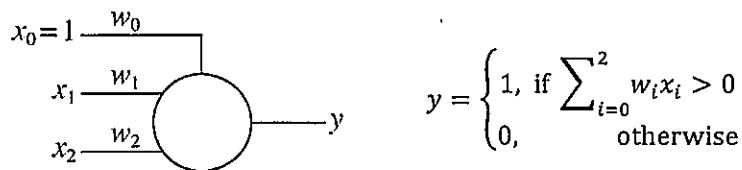


Fig. 3

P. 3. Explain the following terms briefly.

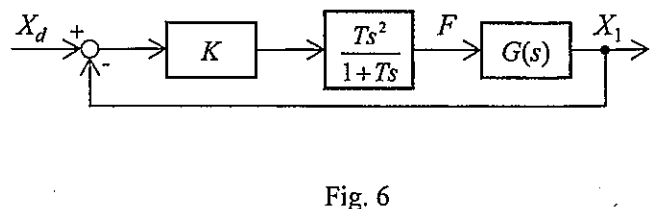
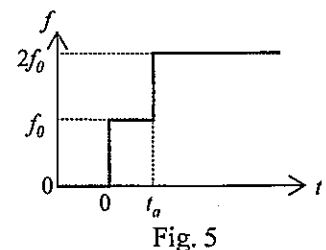
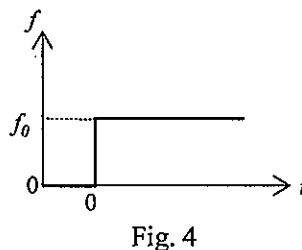
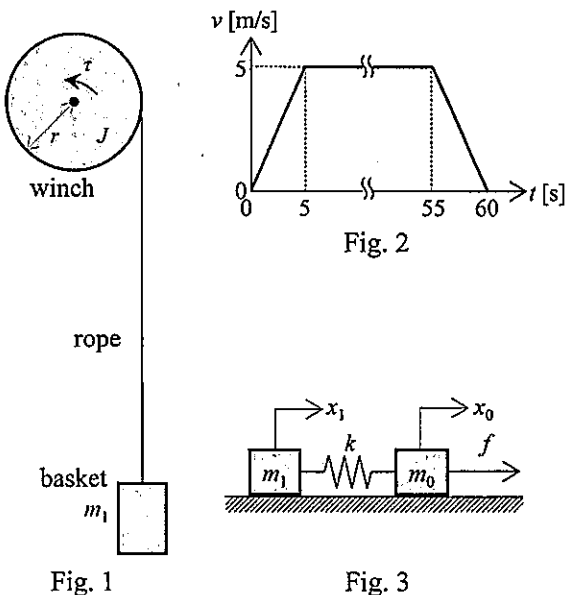
- (1) "process" and "thread" in operating systems
- (2) "Gaussian filter" in image processing
- (3) "sampling theorem" in digital signal processing

MEMO
(Do not detach this page)

Problem 2A (Required Elective)

Consider an elevating machine in Fig. 1 consisting of a winch, rope and basket. Assume that the winch has radius r , moment of inertia J , and drive torque τ , and that mass of the basket is m_1 . Answer the following questions. For simplicity, mass of the rope can be ignored. Also, friction is negligible.

- P. 1. Velocity v [m/s] of the basket is controlled as depicted in Fig. 2. When $J = 100$ [kg m²], $r = 0.80$ [m], $m_1 = 1000$ [kg], calculate the maximum number of revolutions per minute [rpm] and the maximum value of τ [N m] of the winch with 2 significant figures. The acceleration of gravity is 9.8 [m/s²]. Ignore the rope elongation and contraction.
- P. 2. In a motion of an elevating machine in a tall building, elastic effect of the rope becomes non-negligible. To model this effect, assume a system consisting of points of mass with mass m_0 , m_1 and a spring with spring constant k on the horizontal plane as shown in Fig. 3, where force f is applied to m_0 . Let x_0 and x_1 be displacements of m_0 and m_1 from the resting positions, respectively.
- (1) Obtain motion equations of the two-mass system in Fig. 3.
 - (2) The motion equations in the above (1) can be expressed as $\Delta\ddot{x} + \omega_0^2 \Delta x = f/m_0$, where $\Delta x (=x_0 - x_1)$ is a rope elongation. Obtain ω_0 in this equation. In problems below (including P.3.), ω_0 may be used.
 - (3) As shown in Fig. 4, a step force of $f=f_0$ was applied at time $t = 0$. Derive the time variation of Δx , and illustrate it graphically.
 - (4) As shown in Fig. 5, after a step force of $f=f_0$ was applied at time $t = 0$, an additional step force of $f=2f_0$ was applied at $t = t_a$ in order to eliminate an oscillatory component of Δx . Obtain t_a .
- P. 3. Consider a position control of m_1 in the model of Fig. 3.
- (1) In the above P. 2. (1), with input f and output x_1 , obtain transfer function $G(s) = X_1(s)/F(s)$. $F(s)$ and $X_1(s)$ are the Laplace transformations of $f(t)$ and $x_1(t)$, respectively.
 - (2) For the above $G(s)$, a feedback system was constructed as shown in Fig. 6. X_d is the Laplace transformation of target $x_d(t)$. K and T are constants. Derive the condition for this system to be stable.



MEMO
(Do not detach this page)

Problem 2B (Required Elective)

P. 1. Answer the following questions about programming in C.

- (1) S_n denoting the sum of natural numbers from 1 to n is defined as follows. Describe a function `int S(int n)` that returns S_n as its value in two ways, one with a recursive call and another without it.

$$S_n = S_{n-1} + n, \quad \text{where } S_1 = 1$$

- (2) A problem may occur in the recursive implementation of the function `int S(int n)` in (1) when n increases. Describe the problem and its cause.
- (3) The following program is given. `apply(add, x, y)` returns the sum of x and y , and `apply(mul, x, y)` returns the multiplication of x and y . Fill in the underlined part below.

```
typedef int (*func)(int, int);
int add(int x, int y) { return x + y; }
int mul(int x, int y) { return x * y; }
int apply(func f, int x, int y) { return _____ ; }
```

P. 2. List 1 is a program in Python for learning.

The function `update()` updates w according to the following equation, where w is a weight list, x is an input data list, u is a learning rate, and y is a correct label.

$$w = w + uyx \quad (1)$$

The function `predict()` calculates

$$predict(w, x) = g\left(\sum_{i=0}^{d-1} w_i x_i\right) \quad (2)$$

using the function $g(a) = \begin{cases} +1 & (a \geq 0) \\ -1 & (a < 0) \end{cases}$, where d is the length of the lists w and x .

- (1) Fill in (A) of List 1.
- (2) Write the output of this program when it is executed.
- (3) Define the function `sum()`, which corresponds to the symbol \sum in Equation (2), by filling in (B) of List 1.
- (4) (C) in List 1 corresponds to $\sum_{i=0}^{d-1} w_i x_i$ in Equation (2). Fill in (C) with the shortest possible code using the function `sum()` defined above. If necessary, you may define new functions and/or variables and use them.

P. 3. Assume $xs = [[0,0,1], [0,1,1], [1,0,1], [1,1,1]]$, $ys = [-1,1,1,-1]$ in List 1.

- (1) Describe what happens when this version of the program is executed, along with its reasons.
- (2) Consider improving the program so that it succeeds in learning for this xs , ys . Illustrate the simplest possible neural network for this purpose. Also, explain, in about five lines, how to update its weight list.

```

#!/usr/bin/env python

u = 0.25
def main():
    w = [0.125, 0.125, 0.25]
    o = [0, 0, 0, 0]
    xs = [[0, 0, 1], [0, 1, 1], [1, 0, 1], [1, 1, 1]]
    ys = [-1, -1, -1, 1]
    while o != ys:
        for j in [0, 1, 2, 3]:
            o[j] = predict(w, xs[j])
            if o[j] != ys[j]:
                w = update(w, xs[j], ys[j])
        print(o)
    print(w)

def update(w, x, y):
    (A)
    -----

def sum (f, a, b):
    if a > b:
        return 0
    else:
        return (f(a) + (B) )
    -----

def predict(w, x):
    (C)
    -----

    if a >= 0:
        return 1
    else:
        return -1

main()

```

List 1

MEMO

(Do not detach this page)