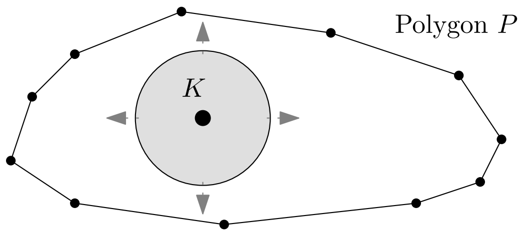
Übungsblatt 6 – Lösungsvorschläge

Vorlesung Algorithmentechnik im WS 09/10

Problem 1: Größter Kreis in konvexem Polygon [vgl. Kapitel 6 im Skript, Vorlesungsfolien vom 7. Jänner 2010]***

Gegeben sei ein konvexes Polygon $P \subseteq \mathbb{R}^2$. Bei dem Problem in dieser Aufgabe suchen wir nach dem größtmöglichen Kreis $K \subseteq \mathbb{R}^2$, der vollständig in P enthalten ist; Das heißt, es gilt $K \subseteq P$ wobei der Radius von K maximal ist.



(a) Geben Sie ein lineares Programm zur Lösung des Problems für ein Polygon P an und erläutern Sie dabei die Zielfunktion, sowie alle Variablen und Nebenbedingungen.

Hinweis: Überlegen Sie sich wie Sie ein konvexes Polygon mit n Seiten definieren können so, dass ausgehend davon die Bedingung $K \subseteq P$ für einen beliebigen Kreis überprüft werden kann. Bauen Sie ausgehend davon Ihr lineares Programm.

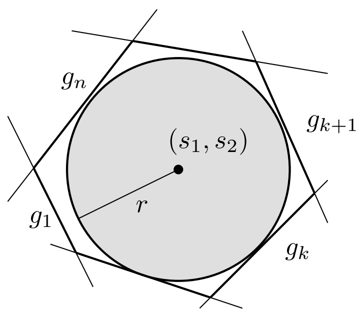
Lösung. Um das lineare Programm zur Lösung unseres Problems zu bestimmen, müssen wir uns zunächst ein paar Gedanken über die geometrische Natur des Problems machen.

Sei als $P \subseteq \mathbb{R}^2$ das gegebene Polygon. Die Anzahl der Seiten des Polygons sei n. Nehmen wir zur Vereinfachung an, dass P keine *vertikalen* Seiten enthält. Man kann sich leicht klarmachen, dass in dem Falle dass mindestens eine Seite vertikal ist, wir durch eine Rotation des Achsenkreuzes (Koordinatentransformation) das Polygon so in eine Position drehen können, dass keine der Seiten mehr vertikal ist. Wegen $n < \infty$ ist dies immer möglich.

Weiterhin definieren wir die Seiten des Polygons durch n Geraden g_i der Form

$$y = a_i x + b_i \qquad i = 1, \ldots, n.$$

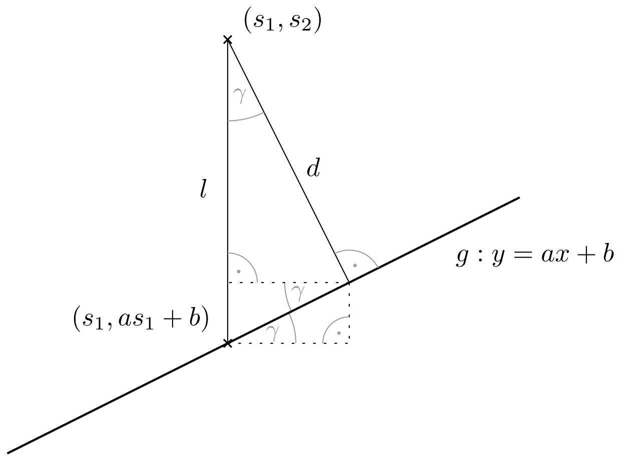
Da wir keine vertikalen Seiten zulassen, sei die Nummerierung der Geraden so, dass $g_1, \ldots, g_k$ das Polygon von unten begrenzen, während $g_{k+1}, \ldots, g_n$ das Polygon von oben begrenzen.



Damit nun der Kreis K mit Mittelpunkt $s := (s_1, s_2)$ und Radius r komplett in P enthalten ist, müssen genau folgende Bedingungen erfüllt sein:

- Der Punkt s hat mindestens den Abstand r von allen Geraden $g_1, \dots, g_n$,
- der Punkt s liegt oberhalb von $g_1, \dots, g_k$ und
- der Punkt s liegt unterhalb von $g_{k+1}, \dots, g_n$.

Zur Berechnung des Abstandes des Punktes s von einer Geraden g, beziehen wir uns auf die folgende Abbildung.



Die Steigung der Geraden g findet sich im Winkel γ wieder über $\tan \gamma = a$. Wegen der Ähnlichkeit der eingezeichneten Dreiecke gilt ebenfalls $\cos \gamma = d/l$, und somit ist der gesuchte Abstand gerade $d = \cos(\gamma) l$. Mit $\gamma = \arctan(a)$ folgt insgesamt $d = \cos(\arctan a) l$. Die Länge der Strecke l ist gerade die Differenz der y-Koordinaten des Kreismittelpunktes (s_1, s_2) und dem Wert $as_1 + b$. Weiterhin ist aus der Analysis bekannt, dass gilt

$$\cos(\arctan x) = \frac{1}{\sqrt{x^2 + 1}},$$

wir erhalten in unserem Fall also insgesamt

$$d = \cos(\arctan a) l = \frac{s_2 - as_1 - b}{\sqrt{a^2 + 1}}.$$

Für den Fall, dass der Punkt (s_1, s_2) unterhalb der Geraden g liegt, gilt analog

$$d = \frac{as_1 + b - s_2}{\sqrt{a^2 + 1}}.$$

Der Kreis K liegt also genau dann vollständig in P wenn folgende Ungleichungen erfüllt sind

$$\begin{aligned} \frac{s_2 - a_i s_1 - b_i}{\sqrt{a_i^2 + 1}} &\geq r, \qquad i = 1, \dots, k \\ \frac{a_i s_1 + b_i - s_2}{\sqrt{a_i^2 + 1}} &\geq r, \qquad i = k+1, \dots, n. \end{aligned}$$

Obwohl man etwas durch den Term $\sqrt{a_i^2 + 1}$ abgeschreckt sein mag, da dieser doch vermeintlich die Linearität der Bedingungen verletzt, stellt dies kein Problem dar, da die a_i Teil der Eingabe sind. Tatsächlich sind also die Terme $\sqrt{a_i^2 + 1}$ Konstanten innerhalb unseres linearen Programms.

Unser Ziel ist es den *größten* Kreis zu finden. Wir definieren uns also ein lineares Programm mit den Variablen r, s_1 und s_2 durch

maximiere r

unter den Nebenbedingungen

$$\frac{s_2 - a_i s_1 - b_i}{\sqrt{a_i^2 + 1}} \geq r, \qquad i = 1, \ldots, k$$
$$\frac{a_i s_1 + b_i - s_2}{\sqrt{a_i^2 + 1}} \geq r, \qquad i = k+1, \ldots, n.$$

Eine Lösung liefert uns demnach einen größtmöglichen Kreis mit Mittelpunkt (s_1, s_2) und Radius r, der vollständig in P enthalten ist.

□

(b) Lässt sich das auf höhere Dimensionen verallgemeinerte Problem (das heißt, wir suchen nach der größtmöglichen Kugel, die in einem konvexen Polytop enthalten ist) ebenfalls durch ein lineares Programm lösen? Begründen Sie Ihre Antwort.

Lösung. Für die Verallgemeinerung auf höhere Dimensionen lässt sich der gleiche Ansatz verwenden. Statt den Punkt-Gerade-Abständen müssen lediglich Punkt-Hyperebenen-Abstände als Bedingungen berechnet werden. Diese sind ebenfalls als lineare Ungleichungen formulierbar.

Im Allgemeinen sei nun H eine Hyperebene für alle $x \in \mathbb{R}^m$ mit $(x - b) \cdot a = 0$, wobei $a, b \in \mathbb{R}^m$ und $a \neq 0$, dann ist der Abstand des Kreismittelpunktes s gegeben durch

$$d = \frac{|(s - b) \cdot a|}{\|a\|_2}.$$

Dabei ist der nicht-lineare Anteil $\|a\|_2$ durch die Eingabe bereits definiert – enthält also insbesondere keine Variablen. Das Skalarprodukt im Zähler führt analog zu Aufgabe (a) wieder zu einer linearen Form. Die Nebenbedingungen (Man muss hier noch darauf achten, dass der Kreismittelpunkt im inneren des Polytops liegt) sind also insgesamt linear, ein linearen Programms führt somit zu dem gewünschten Ergebnis. □

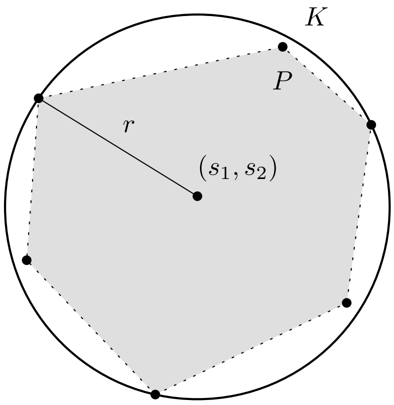
(c) Betrachten Sie das folgende, sehr ähnliche Problem: Zu einem konvexen Polygon $P \subseteq \mathbb{R}^2$ suchen wir den *kleinstmöglichen* Kreis $K \subseteq \mathbb{R}^2$ so, dass P vollständig in K enthalten ist, also $P \subseteq K$ gilt. Können Sie dieses Problem ebenfalls durch ein lineares Programm (ähnlich zu Aufgabe (a)) lösen? Begründen Sie Ihre Antwort.

Lösung. Obwohl das Problem einen möglichst kleinen Kreis K so zu suchen, dass das gegebene Polygon P komlett in K enthalten ist zunächst sehr ähnlich klingt, und man daher vermuten mag, dass man mit der gleichen Strategie zum Ziel kommt, schlägt dies jedoch fehl. In Aufgabe (a) mussten wir lediglich Abstände zwischen dem Kreismittelpunkt und den begrenzenden *Geraden* des Polygons berechnen. Dies war mit einer linearen Formulierung möglich. Wollen wir jedoch, dass umgekehrt, das Polygon im Kreis enthalten ist, so kommen wir mit dieser Modellierung nicht zum Ziel.

Wir betrachten stattdessen folgendes Problem. Zu einer Menge P von *Punkten* (x_i, y_i) suchen wir einen Kreis K mit möglichst kleinem Radius r und Mittelpunkt (s_1, s_2) so, dass alle Punkte $p_i \in P$ ebenfalls in dem Kreis K enthalten sind. Offensichtlich ist dieses Problem äquivalent zu dem Problem dass wir ein Polygon gegeben haben: Wir betrachten als Punktemenge P genau die Ecken des Polygons. Damit das Polygon vollständig in K enthalten ist, müssen die Ecken selbstverständlich auch in K enthalten sein. Des Weiteren ist diese Bedingung bereits hinreichend. Sind alle Ecken des Polygons in K enthalten, so sind es auch die verbindenden Strecken, also die Seiten des Polygons, und damit auch das komplette Polygon selbst.

Somit ist unser Optimierungsziel einen Kreis mit Mittelpunkt $s = (s_1, s_2)$ zu finden so, dass der Radius r des Kreises mindestens so groß wie der Abstand zwischen jedem Punkt $p_i \in P$ und s ist. Formal bedeutet dies:

$$\textbf{minimiere } r$$



unter den Nebenbedingungen

$$\sqrt{(s_1 - x_i)^2 + (s_2 - y_i)^2} \leq r \qquad i = 1, \ldots, n.$$

Bei diesen Ungleichungen sind nun die Variablen des Programms nicht mehr linear. Das heißt, das Problem lässt sich nicht analog zu Aufgabe (a) mit Methoden der linearen Programmierung lösen. □

Problem 2: Euklidischer Geschäftsreisender [vgl. Kapitel 6 im Skript, Vorlesungsfolien vom 7. Jänner 2010] **

Das *Problem des euklidischen Geschäftsreisenden* ist folgendermaßen definiert: Gegeben seien n Punkte $p_i \in P = \{p_1, \ldots, p_n\}$ in der Ebene mit Koordinaten (x_i, y_i) für jeden Punkt $p_i \in P$. Gesucht ist die kürzeste Rundtour, die alle Knoten mindestens einmal besucht. Rundtour bedeutet, dass die Tour an dem gleichen (frei wählbaren) Knoten beginnen und enden muss. Der Abstand zweier Punkte ist definiert durch den euklidischen Abstand der beiden Punkte.

Modellieren Sie das Problem als (gegebenenfalls) ganzahliges lineares Programm. Erklären Sie hierbei Zielfunktion, alle Variablen und alle Nebenbedingungen.

Lösung. Sei $d_{ij} = \|p_j - p_j\| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ vorberechnet. Da für dieses Abstandsmaß die Dreiecksungleichung gilt, ist klar, dass die kürzeste Rundreise jede Stadt nur einmal besucht (es gibt also keine Hubs). Seien x_{ij} binäre Variablen, die angeben, ob die Strecke von p_i nach p_j Teil der Rundreise ist.

- Zielfunktion, minimiere Länge der Rundtour

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} x_{ij}$$

- jeder Punkt hat genau einen Nachfolger

$$\sum_{j=1}^{n} x_{ij} = 1 \; \forall i \in \{1, \ldots, n\}$$

- jeder Punkt hat genau einen Vorgänger

$$\sum_{i=1}^{n} x_{ij} = 1 \; \forall j \in \{1, \ldots, n\}$$

- keine reflexiven Kanten, denn diese sind wegen $d_{ii} = 0$ kostenlos

$$x_{ii} = 0 \; \forall i \in \{1, \ldots, n\}$$

- binäre Variable
$$x_{ij} \in \{0,1\}\ \forall i,j \in \{1,\dots,n\}$$
- keine Subzyklen
$$\forall S \subsetneq \{1,\dots,n\}, S \neq \emptyset : \sum_{i\in S}\sum_{j\notin S} x_{ij} \geq 1$$

Dieses ILP modelliert das Problem zwar korrekt, hat aber wegen der Subzyklenbedingung exponentiell viele Nebenbedingungen. Um diese Anzahl zu reduzieren bedienen wir uns der folgenden Idee. Wir nummerieren die Punkte p_i in der Reihenfolge, in der sie durch die Rundtour besucht werden. Wir führen deshalb $n-1$ weitere Variablen $u_1,\dots,u_{n-1} \in \{1,\dots,n-1\}$ ein. Für zwei Punkte p_i und p_j soll nun gelten $u_i < u_j$, falls es eine Kante von p_i nach p_j in der Tour gibt, also $x_{ij} = 1$ ist. Andernfalls soll über die Ordnung bezüglich u_i und u_j keine Aussage getroffen werden. Dies ist äquivalent zu

$$\begin{aligned} & u_i - M(1-x_{ij}) < u_j \\ \Leftrightarrow\ & u_i - M(1-x_{ij}) \leq u_j - 1 \\ \Leftrightarrow\ & u_i - u_j + Mx_{ij} \leq M - 1 \end{aligned}$$

Für jedes Paar $i,j \in \{1,\dots,n-1\}$ führen wir solch eine Nebenbedingung ein. Dabei ist M eine hinreichend große Konstante, damit für den Fall $x_{i,j} = 0$ die Ungleichung in jedem Fall erfüllt wird. $M := n$ ist dabei hinreichend. Damit ist die Anzahl der Nebenbedingung quadratisch statt exponentiell in n. □

Problem 3: Dualität [vgl. Kapitel 6 im Skript, Vorlesungsfolien vom 7. Jänner 2010] **

Marco Stanley Fogg[1] ist ein Student, dem nach dem Tod seines Onkels lediglich dessen antiquarische Buchsammlung als finanzielle Rücklage bleibt. Um sein Studium möglichst lange durch den Verkauf der Bücher finanzieren zu können, versucht er, seine Ernährung auf ein Minimum zu beschränken. Nachdem er für eine Menge von m wichtigen Nährstoffen $1,\dots,m$ jeweils den minimalen täglichen Bedarf b_i ($i = 1,\dots,m$) für einen Mann seines Alters und Gewichts in Erfahrung gebracht hat, sucht er den lokalen Supermarkt auf und ermittelt für eine Menge von n Produkten $1,\dots,n$ jeweils den Preis pro Einheit c_j ($j = 1,\dots,n$) sowie den Anteil a_{ij} des Nährstoffes i ($i = 1,\dots,m$) am Produkt j ($j = 1,\dots,n$).

(a) Formulieren das beschriebene Optimierungsproblem als lineares Programm L.

Lösung. Das primale lineare Programm ist wie folgt gegeben:

Zielfunktion:

$$\text{minimiere } \sum_{j=1}^{n} c_j x_j \qquad \text{(Kosten des Einkaufs)}$$

Nebenbedingungen:

$$\sum_{j=1}^{n} a_{ij} x_j \geq b_i \qquad i = 1,\dots,m$$

□

[1]Figur aus dem Roman “Moon Palace” von Paul Auster, die Handlung wurde leicht abgeändert ;)

(b) Formulieren Sie das zu L duale lineare Programm D. Geben Sie eine sinnvolle Interpretation des dualen Programms an und überlegen Sie sich dabei, wer ein Interesse daran haben könnte, das duale Programm zu optimieren.

Lösung. Dazu läßt sich das duale Programm wie folgt formulieren:

Zielfunktion:

$$\text{maximiere } \sum_{i=1}^{m} y_i b_i \qquad \text{(Kosten für Tagesbedarf)}$$

Nebenbedingungen:

$$\sum_{i=1}^{m} y_i a_{ij} \leq c_j \qquad j = 1, \dots, n$$

Eine Interpretation des dualen Programms könnte folgendermaßen lauten: Der Vektor y enthält zu jedem Nährstoff i einen Eintrag y_i, der dem festzulegenden Preis einer Einheit des Nährstoffes i entspricht. Das duale Programm maximiert dann die Kosten für einen Einkauf, der den Tagesbedarf von Fogg abdeckt. Dabei werden als Nebenbedingung für jedes Produkt maximale Preise pro Einheit berücksichtigt: Die Kosten der für eine Einheit eines Produktes benötigten Nährstoffe, darf einen maximalen Preis c_i nicht überschreiten. Natürlich möchte der Besitzer des lokalen Supermarktes den Preis eines solchen Einkaufs maximieren. □

Problem 4: Lineare Ungleichungssysteme [vgl. Kapitel 6 im Skript, Vorlesungsfolien vom 7. Jänner 2010] **

In dieser Aufgabe wollen wir zeigen, dass das Problem ein lineares Programm zu lösen äquivalent zu dem Problem ist, zu ein System von Ungleichungen danach zu fragen ob dieses bereits eine gültige Lösung hat.

Gegeben sei also eine Matrix $A \in \mathbb{R}^{m \times n}$ und ein Vektor $b \in \mathbb{R}^m$. Bei dem Problem Linear Inequality Feasibility suchen wir einen Vektor $x \in \mathbb{R}^n$, der die Ungleichungen $Ax \leq b$ erfüllt.

(a) Zeigen Sie, dass mit einem Algorithmus zur Lösung eines linearen Programms auch das Problem Linear Inequality Feasibility gelöst werden kann. Die Anzahl Variablen und Nebenbedingungen für das lineare Programm soll dabei polynomiell in n und m sein.

Lösung. Zu einer gegebenen Instanz (A, b) des Problems Linear Inequality Feasibility definieren wir uns ein lineares Programm wie folgt. Die Matrix A und der Vektor b werden übernommen. Die Zielfunktion wählen wir beliebig, beispielsweise über $c = \vec{1}$. Die Lösung des linearen Programms

$$\textbf{maximiere } c^\top x$$

unter den Nebenbedingungen

$$Ax \leq b$$

führt dann unmittelbar zu einer Belegung der Variablen $x_1, \dots, x_n$, die die Ungleichungen $Ax \leq b$ erfüllen. Ist im Gegensatz das obige lineare Programm unlösbar, so exstiert keine Belegung der Variablen $x_1, \dots, x_n$ die die Ungleichungen $Ax \leq b$ erfüllt. Die Instanz von Linear Inequality Feasibility ist also ebenfalls unlösbar.

Bezüglich der Problemgröße gilt offensichtlich dass die Anzahl Variablen und Nebenbedingungen polynomiell in der Anzahl Variablen und Ungleichungen der Instanz von Linear Inequality Feasibility ist. □

(b) Zeigen Sie die Umkehrung von Aufgabe (a): Gegeben ein lineares Programm, zeigen Sie, wie Sie mit einem Algorithmus für das LINEAR INEQUALITY FEASIBILITY Problem das lineare Programm lösen können. Die Anzahl Variablen und Ungleichungen des Ungleichungssystems soll dabei polynomiell in der Anzahl Variablen und Nebenbedingungen des linearen Programms sein.

Hinweis: Nutzen Sie beim Aufstellen des Ungleichungssystems die Dualitätssätze aus der Vorlesung aus.

Lösung. Wir benutzen zur Lösung der Aufgabe den starken Dualitätssatz aus der Vorlesung. Zu einem gegebenen linearen Programm (**PP**)

$$\textbf{maximiere } x^\top c$$

unter den Nebenbedingungen

$$Ax \leq b$$

konstruieren wir zunächst das duale Programm (**DP**)

$$\textbf{minimiere } y^\top b$$

unter den Nebenbedingungen

$$A^\top y \geq c.$$

Der starke Dualitätssatz aus der Vorlesung besagt nun, dass (**PP**) genau dann lösbar ist, falls (**DP**) ebenfalls lösbar ist, und dass in diesem Falle die Werte der Zielfunktionen zusammenfallen, das heißt für die optimalen Lösung x^* des (**PP**) und die optimalen Lösung y^* des (**DP**) gilt

$${x^*}^\top c = {y^*}^\top b.$$

Wir definieren uns also eine Instanz (B, d) des LINEAR INEQUALITY FEASIBILITY Problems wie folgt. Wir führen $m + n$ Variablen $z_1, \ldots, z_{n+m}$ ein, wobei $z_1, \ldots, z_n$ den Variablen $x_1, \ldots, x_n$ von (**PP**), und $z_{n+1}, \ldots, z_{n+m}$ den Variablen $y_1, \ldots, y_m$ von (**DP**) entsprechen. Die Matrix B und den Vektor d konstruieren wir aus den folgenden Ungleichungen. Die Ungleichungen, die sich aus (**PP**) ergeben

$$A \begin{pmatrix} z_1 \\ \vdots \\ z_n \end{pmatrix} \leq b,$$

den Ungleichungen, die sich aus (**DP**) ergeben

$$A^\top \begin{pmatrix} z_{n+1} \\ \vdots \\ z_{n+m} \end{pmatrix} \geq c,$$

und den Ungleichungen die sich aus dem Wert der Zielfunktion für die Optimallösungen von (**PP**) beziehungsweise (**DP**) ergeben, also

$$\begin{pmatrix} z_1 & \cdots & z_n \end{pmatrix} c = \begin{pmatrix} z_{n+1} & \cdots & z_{n+m} \end{pmatrix} b. \tag{1}$$

Da wegem des schwachen Dualitätssatzes $\begin{pmatrix} z_1 & \cdots & z_n \end{pmatrix} c \leq \begin{pmatrix} z_{n+1} & \cdots & z_{n+m} \end{pmatrix} b$ gilt, genügt es um die Gleichheit in Gleichung (1) zu modellieren, sich auf die Ungleichung $\begin{pmatrix} z_1 & \cdots & z_n \end{pmatrix} c \geq \begin{pmatrix} z_{n+1} & \cdots & z_{n+m} \end{pmatrix} b$ zu beschränken. Das Ungleichungssystem ist also insgesamt definiert durch

$$B := \begin{pmatrix} A & 0 \\ 0 & -A^\top \\ -c^\top & b^\top \end{pmatrix} \qquad \text{und} \qquad d := \begin{pmatrix} b \\ -c \\ 0 \end{pmatrix}$$

Das Ungleichungssystem enthält nun genau dann eine Lösung, falls das lineare Programm lösbar ist. Ist die Optimallösung des (**PP**) eindeutig, so ist auch die zulässige Lösung des Ungleichungssystems eindeutig – andernfalls ist jede Lösung des Systems eine Optimallösung bezüglich (**PP**). Die Werte für die Variablen des (**PP**) ergeben sich aus den Belegungen der Variablen $z_1, \ldots, z_n$; Die Werte für die Variablen des (**DP**) ergeben sich durch $z_{n+1}, \ldots, z_{n+m}$. Weiterhin ist die Größe der Instanz von Linear Inequality Feasibility polynomiell in der Anzahl der Variablen und Ungleichungen von (**PP**).

$\square$