1. Einführung

In diesem Kapitel führen wir zunächst in unsere Problemstellung ein und geben einige (Anwendungs–) Beispiele an. Dann beschäftigen wir uns kurz mit einigen weiteren Problemen der Optimierung, deren Behandlung allerdings nicht Gegenstand dieses Buches ist. Schließlich gehen wir auf einige mögliche Klassifikationen von Optimierungsproblemen ein, die später bei der Auswahl eines geeigneten Verfahrens von Bedeutung sein können.

1.1 Problemstellung und Beispiele

Seien $X \subseteq \mathbb{R}^n$ eine nichtleere Menge und $f : X \to \mathbb{R}$ eine gegebene Funktion. Gegenstand dieses Buches ist das *Minimierungsproblem*

$$\text{minimiere } f(x) \text{ unter der Nebenbedingung } x \in X. \tag{1.1}$$

Hierfür benutzen wir häufig die Kurzschreibweisen

$$\min f(x) \quad \text{u.d.N.} \quad x \in X$$

oder

$$\min_{x \in X} f(x).$$

Die Funktion f wird als *Zielfunktion* von (1.1) bezeichnet, und die Menge X heißt *zulässiger Bereich* oder *zulässige Menge* von (1.1). Ist $X = \mathbb{R}^n$, so spricht man von einem *unrestringierten Minimierungsproblem*, anderenfalls von einem *restringierten Minimierungsproblem.*

Alternativ könnte man auch ein *Maximierungsproblem*

$$\text{maximiere } f(x) \quad \text{u.d.N.} \quad x \in X \tag{1.2}$$

betrachten. In beiden Fällen spricht man von einem *Optimierungsproblem.*

Definition 1.1. *Betrachte das Minimierungsproblem (1.1).*

(i) Ein Vektor $x \in \mathbb{R}^n$ heißt zulässig *für (1.1), wenn $x \in X$ gilt.*

(ii) Ein zulässiger Vektor x^* heißt globales Minimum *von (1.1), wenn*

$$f(x^*) \leq f(x) \quad \textit{für alle } x \in X$$

gilt.

(iii) Ein zulässiger Vektor x^* heißt lokales Minimum *von (1.1), wenn es ein $\varepsilon > 0$ gibt mit*

$$f(x^*) \leq f(x) \quad \textit{für alle } x \in X \cap \mathcal{U}_\varepsilon(x^*);$$

dabei bezeichnet $\mathcal{U}_\varepsilon(x^*) := \{x \in \mathbb{R}^n \mid \|x - x^*\| < \varepsilon\}$ die ε–Kugel um den Punkt x^*.

(iv) Ein zulässiger Vektor x^* heißt striktes globales Minimum *von (1.1), wenn*

$$f(x^*) < f(x) \quad \textit{für alle } x \in X \textit{ mit } x \neq x^*$$

gilt.

(v) Ein zulässiger Vektor x^* heißt striktes lokales Minimum *von (1.1), wenn es ein $\varepsilon > 0$ gibt mit*

$$f(x^*) < f(x) \quad \textit{für alle } x \in X \cap \mathcal{U}_\varepsilon(x^*) \textit{ mit } x \neq x^*.$$

Entsprechend lässt sich ein (striktes) lokales oder globales Maximum für das Optimierungsproblem (1.2) definieren. Da x^* aber offenbar genau dann ein lokales (globales) Maximum von (1.2) ist, wenn x^* ein lokales (globales) Minimum von

$$\min -f(x) \quad \text{u.d.N.} \quad x \in X$$

ist, können (und werden) wir uns bei der Untersuchung von Optimierungsproblemen o.B.d.A. auf Minimierungsprobleme beschränken.

Probleme mit $n = 1$ oder $n = 2$ Variablen lassen sich gut veranschaulichen. Für $n = 1$ kann man den Graphen der Zielfunktion f betrachten. Für die in Abbildung 1.1 gezeichnete Funktion f auf $X = [x_1, x_8]$ ist x_2 das (strikte) globale Minimum, x_7 das (strikte) globale Maximum, x_1 und x_3 sind strikte lokale Maxima, x_8 ist ein striktes lokales Minimum und alle Punkte des Intervalls $[x_4, x_5]$ sind (nicht strikte) lokale Minima (die inneren Punkte dieses Intervalls sind zugleich lokale Maxima); x_6 ist weder ein lokales Minimum noch ein lokales Maximum.

Auch im Fall $n = 2$ kann man unter Verwendung eines Zeichenprogrammes (wie es etwa im MATLAB–Paket enthalten ist) den Graphen einer Funktion f zeichnen lassen. Beispielsweise wird in Abbildung 1.2 der Graph der Funktion

$$f(x, y) := \exp(-x^2 - y^2) + 4\sin(x)\sin(y)/(x^2 + y^2 + 1) \tag{1.3}$$

wiedergegeben; es ist aber schon nicht mehr ganz leicht, aus dieser Abbildung die genaue Lage eines (lokalen oder globalen) Minimums abzulesen.

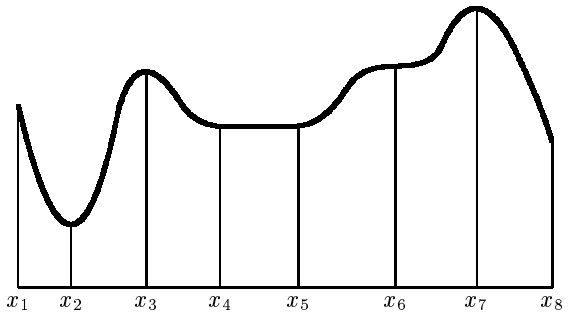

Abb. 1.1. Minima und Maxima einer Funktion einer Variablen

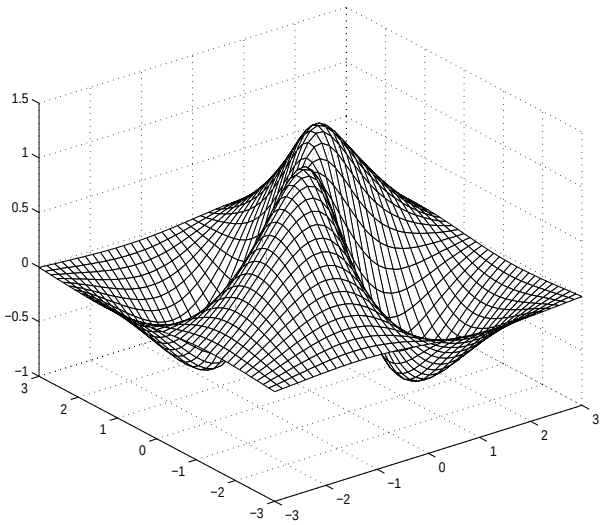

Abb. 1.2. Graph einer Funktion mit zwei Variablen

Eine alternative (und häufig übersichtlichere, wenngleich weniger „spektakuläre“) Möglichkeit zur Veranschaulichung einer Funktion mit $n = 2$ Variablen besteht darin, sich die sogenannten *Höhenlinien* (auch *Niveaulinien* genannt)

$$H_c := \{x \in X \mid f(x) = c\}$$

für verschiedene Werte von c anzusehen. Die Abbildung 1.3 zeigt beispielsweise einige Höhenlinien der Funktion aus (1.3).

Wir kehren nun zu dem Minimierungsproblem (1.1) mit beliebigem $n \geq 1$ zurück. Ein solches Minimierungsproblem liegt häufig in der folgenden Gestalt vor:

$$\min f(x) \quad \text{u.d.N.} \quad g(x) \leq 0,\ h(x) = 0, \tag{1.4}$$

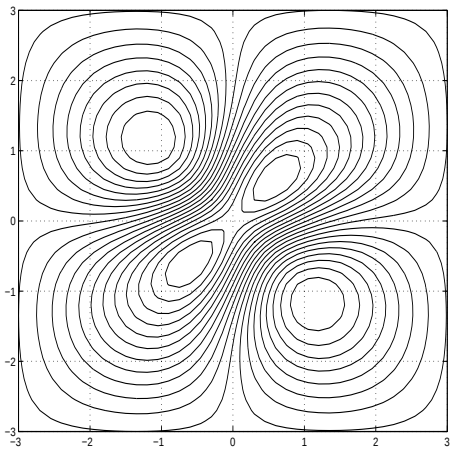

Abb. 1.3. Höhenlinien einer Funktion mit zwei Variablen

wobei $g : \mathbb{R}^n \to \mathbb{R}^m$ und $h : \mathbb{R}^n \to \mathbb{R}^p$ gegebene Funktionen sind und die Nebenbedingung $g(x) \leq 0$ komponentenweise zu verstehen ist:

$$g(x) \leq 0 \;:\Longleftrightarrow\; g_i(x) \leq 0 \quad \text{für alle } i = 1, \ldots, m.$$

Bei dem Minimierungsproblem (1.4) ist der zulässige Bereich X also gegeben durch

$$X = \{x \in \mathbb{R}^n \mid g(x) \leq 0,\, h(x) = 0\}.$$

Das so einfach zu formulierende Minimierungsproblem (1.1) bzw. (1.4) erhält seine Bedeutung daraus, dass es ein *mathematisches Modell* für viele reale Probleme (aus den Ingenieurwissenschaften, Physik, Chemie, Medizin, Ökonomie, Ökologie usw.) ist. Dabei stelle man sich unter den Komponenten von $x \in \mathbb{R}^n$ Parameter des Modells vor, die durch die Modellbildung noch nicht festgelegt sind oder die Entscheidungsfreiheiten beschreiben. Diese Parameter sollen nun optimal gewählt werden. Wir illustrieren dies anhand einiger Beispiele, die bewusst einfach gehalten sind, um einige charakteristische Eigenschaften gut erkennen zu können. Man muss sich aber im Klaren darüber sein, dass die Modellierung realer Probleme schon für sich eine schwierige Aufgabe sein kann und die resultierenden mathematischen Optimierungsprobleme häufig sehr komplex sind. Eine ganze Reihe von Optimierungsproblemen aus verschiedenen Anwendungsfeldern und eine Fülle diesbezüglicher Literaturhinweise findet man beispielsweise in [10], vergleiche auch [5, 6, 21, 113].

Beispiel 1.2. (Angebotsauswertung [21])
Ein Unternehmen will eine bestimmte Menge M eines Gutes einkaufen und holt dazu Angebote von n Lieferfirmen ein, von denen keine die gewünschte

Gesamtmenge alleine liefern kann. Das Angebot des Anbieters i enthält maximale Liefermengen $\max_i$ und Preise $f_i(x_i)$ als Funktion der bestellten Menge x_i. Die Funktionen f_i sind in der Regel monoton steigend und möglicherweise nichtlinear. Die Aufgabe besteht darin, die bei jedem Anbieter zu bestellenden Mengen so zu bestimmen, dass die gesamten Einkaufskosten möglichst klein werden. Als Optimierungsproblem ergibt sich somit

$$\begin{array}{ll} \min & f(x) := \sum_{i=1}^n f_i(x_i) \\ \text{u.d.N.} & \sum_{i=1}^n x_i = M, \\ & 0 \le x_i \le \max_i \quad \text{für alle } i = 1, 2, \ldots, n. \end{array}$$

Beispiel 1.3. (Produktions–Lagerhaltungsproblem)
In einem Betrieb wird ein Produkt P während n Perioden produziert, etwa x_j Einheiten während der Periode j, $j = 1, \ldots, n$. Das Produkt kann bis zu einer Menge von L Einheiten gelagert werden. In Periode j ist (bis zum Ende dieser Periode) ein bestimmter Bedarf b_j zu befriedigen; dazu kann auf die laufende Produktion oder auf den zu Beginn dieser Periode vorhandenen Lagerbestand y_j zurückgegriffen werden, $j = 1, \ldots, n$. Zu Beginn der Periode 1 sind s_0 Einheiten vorhanden, am Ende der Periode n sollen (für Reklamationen oder Nachbestellungen) noch s_1 Einheiten vorrätig sein.

Für die Abhängigkeit der Produktionskosten in Periode j von der produzierten Menge liegen Erfahrungswerte in Form einer (monoton wachsenden und nur bei starker Vereinfachung als linear anzunehmenden) Funktion $f_j(x_j)$ vor. Weiter fallen zu Beginn jeder Periode für den zu diesem Zeitpunkt vorhandenen Lagerbestand Lagerkosten in Höhe von K DM pro Einheit an.

Die Produktion ist unter Beachtung aller genannten Auflagen so zu steuern, dass die Summe der Produktions– und Lagerkosten über alle Perioden möglichst klein ausfällt. Dies führt auf ein Optimierungsproblem in den Variablen x und y von der Gestalt

$$\begin{array}{ll} \min & F(x,y) := \sum_{j=1}^n f_j(x_j) + K \sum_{j=1}^n y_j \\ \text{u.d.N.} & x_j \ge 0, \quad j = 1, \ldots, n, \\ & y_j \le L, \quad j = 1, \ldots, n, \\ & x_j + y_j \ge b_j, \quad j = 1, \ldots, n, \\ & y_{j+1} = y_j + x_j - b_j, \quad j = 1, \ldots, n, \\ & y_1 = s_0, \\ & y_n + x_n - b_n = s_1. \end{array}$$

(Die Vorzeichenbedingungen $y_j \ge 0$ brauchen nicht explizit gestellt zu werden, da sie sich aus den übrigen Nebenbedingungen ergeben.)

Aus der Beziehung $y_1 = s_0$ und der Rekursion $y_{j+1} = y_j + x_j - b_j$ ergibt sich

$$y_{j+1} = s_0 + \sum_{i=1}^{j} (x_i - b_i), \quad j = 1, \ldots, n-1.$$

Einsetzen liefert dann das folgende Minimierungsproblem, welches nur noch von den Variablen x_j abhängt:

$$\begin{array}{ll} \min & f(x) := \sum_{j=1}^{n} f_j(x_j) + K\left(ns_0 + \sum_{i=1}^{n-1}(n-i)(x_i - b_i)\right) \\ \text{u.d.N.} & s_0 + \sum_{i=1}^{j}(x_i - b_i) \le L, \quad j = 1, \ldots, n-1, \\ & s_0 + \sum_{i=1}^{j}(x_i - b_i) \ge 0, \quad j = 1, \ldots, n-1, \\ & s_0 + \sum_{i=1}^{n}(x_i - b_i) = s_1, \\ & x_j \ge 0, \quad j = 1, \ldots, n. \end{array}$$

Beispiel 1.4. (Frachtproblem [92])
Es sollen $5.000\,m^3$ einer Ware innerhalb eines Planungszeitraumes vom Produzenten zu einem Kunden gebracht werden. Die Ware wird in gleichen quaderförmigen Behältern der Höhe x_1, Breite x_2 und Länge x_3 (in m) transportiert, deren Volumen höchstens 1 m^3 ist und die beim Kunden verbleiben.

Das Material für Boden und die vier Seiten der Behälter kostet $4,00$ DM pro m^2. Die Deckel können aus einem Material hergestellt werden, das $0,50$ DM pro m^2 kostet, von dem im Planungszeitraum aber nur $6.500\,m^2$ erhältlich sind. Die Frachtkosten betragen 50 DM für jeden Behälter.

Die Frage ist jetzt, wie die Behälter zu bemessen sind, um die Gesamtkosten möglichst gering zu halten.

Die Anzahl n der Behälter ist (ohne Ganzzahligkeitsforderung an n)

$$n = \frac{\text{Gesamtvolumen}}{\text{Volumen eines Behälters}} = \frac{5.000}{x_1 x_2 x_3}. \tag{1.5}$$

Die Gesamtkosten belaufen sich daher auf

$$f(x) := 50n + 4n(2x_1x_2 + 2x_1x_3 + x_2x_3) + 0.5nx_2x_3.$$

Einsetzen von (1.5) liefert die Optimierungsaufgabe

$$\begin{array}{ll} \min & f(x) := \frac{250.000}{x_1x_2x_3} + \frac{22.500}{x_1} + \frac{40.000}{x_2} + \frac{40.000}{x_3} \\ \text{u.d.N.} & x_1x_2x_3 \le 1, \quad \frac{5.000}{x_1} \le 6.500, \\ & x_i > 0 \quad \text{für alle } i = 1, 2, 3. \end{array}$$

Man beachte, dass im letztgenannten Beispiel strikte Ungleichungen auftreten; durch geschickte Transformation ist es häufig möglich, solche Ungleichungen zu beseitigen, was man unter dem Stichwort *geometrische Optimierung* etwa in [57] nachlesen kann.

Beispiel 1.5. (Ein Parameteridentifizierungsproblem aus der Pharmakokinetik [18])
Für die Wirksamkeit einer medikamentösen Therapie (z.B. mit einem Antibiotikum) ist es wichtig, dass ein definierter Blutspiegel während eines bestimmten Zeitraums aufrechterhalten wird. Um dies steuern zu können, ist es wünschenswert, die Abhängigkeit der Blutspiegelkurve von bestimmten pharmakokinetischen Bestimmungsgrößen zu kennen.

Für die Aufstellung eines mathematischen Modells für den Auf– bzw. Abbau des Blutspiegels $y(t)$ eines oral eingenommenen Medikaments unterscheidet man zwischen dem Invasionsvorgang (die Invasionsgeschwindigkeit

ist proportional zum unresorbierten Anteil des Medikaments) und dem Eliminationsvorgang (die Eliminationsgeschwindigkeit ist proportional zum jeweils vorhandenen Blutspiegel). Bezeichnet $y_I(t)$ den Invasionsanteil des Blutspiegels zur Zeit t (d.h., ohne Berücksichtigung des gleichzeitig ablaufenden Eliminationsvorgangs), so ist

$$\frac{dy_I(t)}{dt} = k_1(a - y_I(t))$$

(k_1 Invasionskonstante, a Sättigungswert). Entsprechend gilt für den Eliminationsanteil $y_E(t)$ mit $y(t) = y_I(t) - y_E(t)$

$$\frac{dy_E(t)}{dt} = k_2 y(t)$$

(k_2 = Eliminationskonstante). Aus diesen Differentialgleichungen und den Anfangsbedingungen $y_I(0) = 0$ und $y(0) = 0$ gewinnt man mittels einer einfachen Rechnung die sogenannte *Bateman–Funktion*

$$y(t; k_1, k_2, a) = \begin{cases} \frac{ak_1}{k_2 - k_1}(e^{-k_1 t} - e^{-k_2 t}), & \text{falls} \quad k_1 \neq k_2, \\ ak_1 t e^{-k_1 t}, & \text{falls} \quad k_1 = k_2. \end{cases}$$

Sinnvoll sind hierbei nur Parameter k_1, k_2, a mit $l_1 \leq k_1 \leq u_1$, $l_2 \leq k_2 \leq u_2$, $a \geq 0$.

Soviel zur Modellierung, und nun zur eigentlichen Aufgabe: Zu gegebenen Messdaten (t_i, y_i), $i = 1, \ldots, m$, mit $m \gg 3$ sind die Parameter k_1, k_2, a so zu bestimmen, dass gilt

$$y(t_i; k_1, k_2, a) \approx y_i, \quad i = 1, \ldots, m,$$

wobei $\approx$ eine „möglichst gute“ Übereinstimmung bezeichnet. Die übliche, angesichts der statistischen Natur der Daten auch vernünftige Präzisierung von $\approx$ („Methode der kleinsten Quadrate“) führt auf die Optimierungsaufgabe

$$\begin{array}{ll} \min & \sum_{i=1}^{m} \left(y(t_i; k_1, k_2, a) - y_i\right)^2 \\ \text{u.d.N.} & l_1 \leq k_1 \leq u_1,\ l_2 \leq k_2 \leq u_2,\ a \geq 0. \end{array} \tag{1.6}$$

Beispiel 1.6. (Entwurf eines Kurbelgetriebes [168])
Mittels eines Kurbelgetriebes soll eine gleichförmige Drehbewegung in eine Schwingbewegung mit vorgegebenem zeitlichem Ablauf umgesetzt werden. Das Kurbelgetriebe bestehe aus einem Gelenkviereck mit Stäben der Längen $\ell_1, \ell_2, \ell_3, \ell_4$ (siehe Abbildung 1.4). Die Eckpunkte $A = (0, 0)$ und $B = (\ell_4, 0)$ seien fest verankert, der Eckpunkt P laufe mit konstanter Geschwindigkeit auf einem Kreis um A ($P = (\ell_1 \cos t, \ell_1 \sin t)$, wobei t der in der Abbildung 1.4 eingezeichnete Winkel bei A ist). Demzufolge läuft der Punkt Q auf einem Abschnitt des Kreises vom Radius ℓ_3 um B, wobei sich die Entfernung u der Punkte P und B ebenso wie die eingezeichneten Winkel w_1, w_2 und w beim Punkt B mit t ändern.

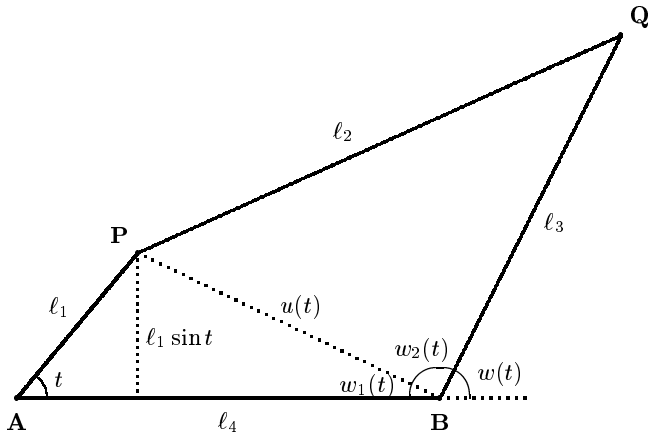

Abb. 1.4. Kurbelgetriebe

Mittels einer einfachen trigonometrischen Rechnung (unter Verwendung des Cosinus–Satzes) bestätigt man die folgenden Gleichungen:

$$\begin{aligned} \tan w_1(t) &= \frac{\ell_1 \sin t}{\ell_4 - \ell_1 \cos t}, \\ (u(t))^2 &= \ell_1^2 + \ell_4^2 - 2\ell_1\ell_4 \cos t, \\ \ell_2^2 &= (u(t))^2 + \ell_3^2 - 2u(t)\ell_3 \cos w_2(t). \end{aligned}$$

Hieraus erhält man $w_1(t)$, $w_2(t)$ und damit eine Darstellung für den die Bewegung des Punktes Q beschreibenden Winkel $w(t)$:

$$w(t) = w(t; \ell_1, \ell_2, \ell_3, \ell_4) = \pi - w_1(t) - w_2(t).$$

Die Aufgabe besteht nun darin, die Längen ℓ_1, ℓ_2, ℓ_3, ℓ_4 der vier Stäbe so festzulegen, dass die Funktion $w(\,\cdot\,; \ell_1, \ell_2, \ell_3, \ell_4)$ möglichst gut mit einer vorgegebenen Funktion $v(\,\cdot\,)$ übereinstimmt. Sinnvoll ist hierbei die Forderung

$$\min \quad \max_{t \in [0, 2\pi]} |v(t) - w(t; \ell_1, \ell_2, \ell_3, \ell_4)|. \tag{1.7}$$

Allerdings unterliegen die Längen ℓ_1, ℓ_2, ℓ_3, ℓ_4 aus technischen Gründen bestimmten Einschränkungen. Wäre z.B. $\ell_2 + \ell_3 < \ell_1 + \ell_4$, so könnte der Punkt P gar keine volle Kreisbewegung ausführen (beachte die Lage mit $t = \pi$). Wir führen exemplarisch einige leicht einsehbare Restriktionen auf ($\ell_{\min}$ sei eine vorgegebene positive Zahl):

(i) $\ell_1 \geq \ell_{\min}$, $\ell_2 \geq \ell_{\min}$,
(ii) $\ell_4 \geq \ell_1 + \ell_{\min}$,
(iii) $\ell_2 + \ell_3 \geq \ell_1 + \ell_4$,
(iv) $|\ell_2 - \ell_3| \leq \ell_4 - \ell_1$ (bzw. $(\ell_2 - \ell_3)^2 \leq (\ell_4 - \ell_1)^2$)

(zu (iv) beachte man die Lage $t = 0$ und verwende, dass die Länge jeder Seite des dann entstehenden Dreiecks PBQ nicht größer als die Summe der Längen der jeweils anderen beiden Seiten sein kann, woraus sich insbesondere die Forderungen $\ell_2 \leq \ell_3 + (\ell_4 - \ell_1)$ und $\ell_3 \leq \ell_2 + (\ell_4 - \ell_1)$ ergeben, die zusammen gerade (iv) liefern).

Leider ist die Zielfunktion in (1.7) auf Grund der Maximumbildung im Allgemeinen nicht überall differenzierbar. Um die darauf beruhenden Schwierigkeiten zu umschiffen, kann man folgendermaßen vorgehen: Man bestimmt das Maximum nicht auf dem ganzen Intervall $[0, 2\pi]$, sondern auf einem hinreichend feinen Raster $\{t_1, t_2, \ldots, t_K\} \subseteq [0, 2\pi]$ und bringt dann das Problem durch Einführen einer neuen Variablen Δ mittels einer kleinen Rechnung auf die Form

$$\begin{array}{ll} \min & \Delta \\ \text{u.d.N.} & -\Delta \leq v(t_k) - w(t_k; \ell_1, \ell_2, \ell_3, \ell_4) \leq \Delta, \quad k = 1, 2, \ldots, K, \\ & \text{(i)} - \text{(iv)}. \end{array} \tag{1.8}$$

Auch dies ist ein Optimierungsproblem der Form (1.4) (mit dem Variablenvektor $x = (\ell_1, \ell_2, \ell_3, \ell_4, \Delta)$).

Abschließend betrachten wir noch ein weiteres Beispiel, das uns im nächsten Abschnitt insbesondere dazu dienen wird, die von uns im Rahmen dieses Buches zu untersuchenden Optimierungsprobleme von einigen weiteren Minimierungsaufgaben abzugrenzen.

Beispiel 1.7. (Design der Nase eines Flugzeugs [67])
Zu entwerfen ist die Nase eines Flugzeugs mit dem Ziel, seinen Luftwiderstand bei einer vorgegebenen Reisegeschwindigkeit zu minimieren. Dazu muss die Nase durch endlich viele Systemparameter beschrieben werden; man kann etwa die Nase aus vier Kegelstümpfen und einem Kugelabschnitt zusammensetzen, siehe Abbildung 1.5.
Die Flugzeugnase ist dann bei vorgegebenem Endradius R durch die Radien r_1, r_2, r_3, r_4, die Winkel $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ und den Kugelradius r_0 vollständig beschrieben. Weiter ist der Luftwiderstand als Funktion des Parametervektors

$$x = (r_0, \ldots, r_4, \alpha_1, \ldots, \alpha_4)^T \in \mathbb{R}^9$$

zu formulieren:

$$LW = LW(r_0, r_1, r_2, r_3, r_4, \alpha_1, \alpha_2, \alpha_3, \alpha_4).$$

Hinzu kommen nun Restriktionen, die technischer, ökonomischer oder ästhetischer Art sein können, wie z.B. bei vorgegebenem Mindestvolumen V_0 und vorgegebener Höchstlänge L_0

$$\begin{array}{ll} \text{(i)} & \text{Volumen}(r_0, \ldots, \alpha_4) \geq V_0, \\ \text{(ii)} & \text{Länge}(r_0, \ldots, \alpha_4) \leq L_0, \\ \text{(iii)} & 0 \leq r_i \leq R, \quad i = 0, 1, \ldots, 4, \\ \text{(iv)} & 0 \leq \alpha_4 \leq \alpha_3 \leq \alpha_2 \leq \alpha_1 \leq \frac{\pi}{2}. \end{array}$$

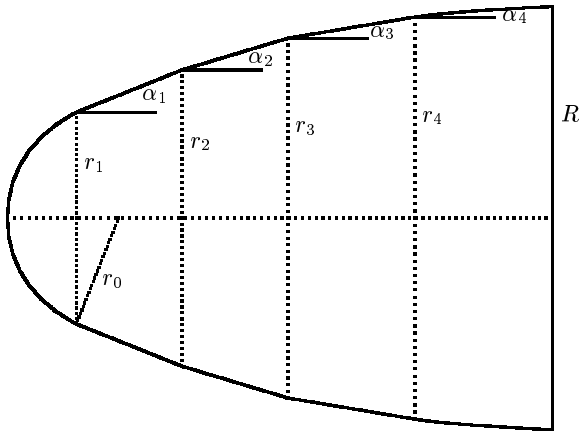

Abb. 1.5. Nase eines Flugzeugs

Weiter wird die mathematische Beschreibung des Luftwiderstandes nur in einem bestimmten Parameterbereich gültig sein, der etwa durch

$$\text{(v)}\ g_i(r_0, \ldots, \alpha_4) \leq G_i, \quad i = 1, \ldots, m,$$

erfasst werden könnte. Zusammengenommen erhält man das Minimierungsproblem

$$\min LW(r_0, \ldots, \alpha_4) \quad \text{u.d.N.} \quad \text{(i)} - \text{(v)}.$$

Die zulässige Menge X wird hier also durch die Ungleichungen (i) – (v) beschrieben (und ist hoffentlich nichtleer!).

1.2 Abgrenzung

Nachdem wir im vorigen Abschnitt die in diesem Buch zu untersuchende Problemstellung (1.1) formuliert haben, wollen wir hier andeuten, dass es neben (1.1) durchaus weitere Optimierungsprobleme gibt, die wir aber nicht näher untersuchen werden. Dazu kehren wir noch einmal zu dem Beispiel 1.7 zurück.

Beispiel 1.8. Zu einem feineren Modell als in Beispiel 1.7 kommt man, wenn man die Flugzeugnase als Rotationskörper einer Funktion $r(\ell), \ell \in [0, L]$, mit $r(0) = 0$ und $r(L) = R$ beschreibt, siehe Abbildung 1.6.
Der Luftwiderstand ist jetzt in der Form $LW(r)$ zu beschreiben, die Restriktionen können etwa lauten

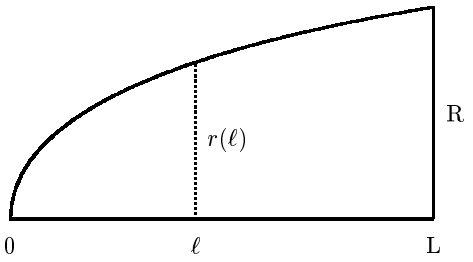

Abb. 1.6. Nase eines Flugzeugs als Rotationskörper

(i') $\pi \int_0^L r^2(\ell) d\ell \geq V_0$,
(ii') $L \leq L_0$,
(iii') $0 \leq r(\ell) \leq R, \ \ \ell \in [0, L]$,
(iv') $r'(\ell) \geq 0, \ \ r''(\ell) \leq 0, \ \ \ell \in [0, L]$.

Zur Formulierung der letztgenannten Bedingungen (die besagen, dass r monoton wächst und konkav ist) muss natürlich vorausgesetzt werden, dass die Funktion r zumindest zweimal stetig differenzierbar ist.

Im Beispiel 1.8 ist der zugrundeliegende Raum der zweimal stetig differenzierbaren Funktionen unendlichdimensional. Minimierungsprobleme in unendlichdimensionalen Räumen werden häufig als *infinite Optimierungsprobleme* bezeichnet, siehe etwa [4, 120]. Diese treten insbesondere in der *Variationsrechnung* sowie bei *Kontroll– oder Steuerungsproblemen* auf. Bei den letztgenannten Problemen treten als Nebenbedingungen auch Differentialgleichungen auf, siehe [81, 88, 23, 52, 90].

Beschränkt man sich im Beispiel 1.8 bei der Wahl der Funktion r auf einen endlichdimensionalen Raum (etwa einen durch endlich viele Funktionen aufgespannten Teilraum der zweimal stetig differenzierbaren Funktionen), so hat man immer noch unendlich viele Restriktionen zu erfüllen. Man spricht dann von einem *semi–infiniten Optimierungsproblem*, vgl. [68, 84, 83]. Semi–infinite Probleme lassen sich zwar in der Form (1.1) schreiben, sie ordnen sich aber im Allgemeinen nicht der Problemstellung (1.4) unter. Durch geeignete *Diskretisierung* von infiniten und semi–infiniten Problemen kommt man jedoch wieder auf Optimierungsprobleme der Gestalt (1.4) (und zwar zumeist auf solche von sehr großer Dimension)! Beispielsweise erhält man im Beispiel 1.6, wenn man eine Umformung in eine zu (1.8) analoge Form vornimmt, ein semi–infinites Optimierungsproblem, (1.8) selbst ist dann eine zugehörige Diskretisierung.

Eine weitere Problemklasse, die in diesem Buch unberücksichtigt bleiben soll, besteht aus Aufgaben, bei denen die Parameter (oder einige davon) auf diskrete Punktmengen beschränkt sind, etwa weil die x_i Mengengrößen

bedeuten, welche nur in ganzen Einheiten sinnvoll sind (relevant ist die Ganzzahligkeitsforderung in der Regel allerdings nur bei kleinen Werten x_i) oder weil die x_i Null–Eins–Entscheidungen beschreiben. Solche Aufgaben der *ganzzahligen* oder *kombinatorischen Optimierung* sind trotz der häufig gegebenen Endlichkeit des zulässigen Bereiches X alles andere als trivial; sie erfordern zumeist jedoch Methoden, die von denen der hier zu diskutierenden *kontinuierlichen Optimierung* völlig verschieden sind, siehe etwa [144, 133, 14]. Eine ganze Reihe von Verfahren zur Lösung ganzzahliger Probleme basieren jedoch auch auf der Lösung einer endlichen Anzahl kontinuierlicher Minimierungsprobleme der Form (1.1) bzw. (1.4), so dass die später zu beschreibenden Verfahren durchaus auch als Grundlage der ganzzahligen Optimierung dienen können.

Schließlich möchte man manchmal auch bezüglich mehrerer Ziele gleichzeitig minimieren. Dies ist der Inhalt der sogenannten *Vektoroptimierung*. Unter Verwendung einer geeigneten Gewichtung der verschiedenen Zielfunktionen erhält man aber gerade wieder ein Minimierungsproblem der Gestalt (1.1).

1.3 Klassifikation

Bei der Lösung von Optimierungsproblemen der Gestalt (1.1) bzw. (1.4) wird man zunächst die Struktur eines vorliegenden Problemes untersuchen, um anschließend ein geeignetes Lösungsverfahren auszuwählen. Wir listen im Folgenden einige Unterscheidungsmerkmale auf (siehe auch [67]), die zum Teil auch für die Gliederung dieses Buches wichtig sind:

(a) Das Minimierungsproblem (1.4) hat
   - α) keine Restriktionen,
   - β) nur Gleichungsrestriktionen ($m = 0$),
   - γ) nur Ungleichungsrestriktionen ($p = 0$),
   - δ) Gleichungs– und Ungleichungsrestriktionen.

   Das vorliegende Buch behandelt im Wesentlichen restringierte Probleme. Eine ausführliche Darstellung von Methoden der *unrestringierten Optimierung* findet man in [66].

(b) Probleme der Form (1.4) von besonderer Bedeutung:
   1) *Lineare Optimierung*: f, g und h sind linear (genauer: affin–linear), z.B.
   $$\min c^T x \quad \text{u.d.N.} \quad Ax = b,\ x \geq 0$$
   mit $A \in \mathbb{R}^{p \times n}, c \in \mathbb{R}^n, b \in \mathbb{R}^p$. Lineare Optimierungsprobleme werden häufig auch als *lineare Programme* bezeichnet, vergleiche Kapitel 3.
   2) *Quadratische Optimierung*: f ist quadratisch und g, h sind linear, z.B.

$$\min \frac{1}{2}x^T Qx + c^T x + \gamma \quad \text{u.d.N.} \quad Ax = b,\ x \geq 0$$

mit $Q \in \mathbb{R}^{n\times n}$ symmetrisch, $A \in \mathbb{R}^{p\times n}, c \in \mathbb{R}^n, b \in \mathbb{R}^p, \gamma \in \mathbb{R}$.

3) *Optimierung mit Box–Restriktionen*: f ist beliebig nichtlinear und die Variablen x_i liegen zwischen gewissen unteren und oberen Schranken:

$$\min f(x) \quad \text{u.d.N.} \quad l_i \leq x_i \leq u_i \quad \text{für alle } i = 1, \ldots, n$$

mit $f : \mathbb{R}^n \to \mathbb{R}, l_i \in \mathbb{R} \cup \{-\infty\}, u_i \in \mathbb{R} \cup \{+\infty\}$ und $l_i < u_i$ für alle $i = 1, \ldots, n$.

4) *Linear restringierte Optimierung*: f ist nichtlinear, und g, h sind linear, z.B.

$$\min f(x) \quad \text{u.d.N.} \quad Ax = b,\ x \geq 0$$

mit $f : \mathbb{R}^n \to \mathbb{R}, A \in \mathbb{R}^{p\times n}, b \in \mathbb{R}^p$.

5) *Konvexe Optimierung*: f ist konvex (siehe Abschnitt 2.1), h ist linear und alle Komponentenfunktionen g_i von g sind konvex:

$$\min f(x) \quad \text{u.d.N.} \quad Ax = b,\ g(x) \leq 0$$

mit $f, g_i : \mathbb{R}^n \to \mathbb{R}$ konvex $(i = 1, \ldots, m), A \in \mathbb{R}^{p\times n}, b \in \mathbb{R}^p$.

6) *Nichtlineare Optimierung* : f, g und h sind beliebig nichtlinear, also im Allgemeinen weder linear noch konvex etc. Dies ist zweifellos das prinzipiell schwierigste Problem in dieser Auflistung.

(c) Glattheitseigenschaften der beteiligten Funktionen f, g und h:
Sind f, g und h zumindest stetig differenzierbar (möglichst sogar zweimal stetig differenzierbar), so spricht man von einem *glatten Optimierungsproblem*, anderenfalls von einem *nichtglatten Optimierungsproblem*. Bezüglich der letztgenannten Klasse vergleiche man auch die Beispiele im dafür einschlägigen Kapitel 6.

(d) Numerisch verfügbare Glattheit von f, g und h:
Seien f, g und h als (zweimal) stetig differenzierbar vorausgesetzt. Dann existieren zwar die Gradienten (und Hesse–Matrizen) der beteiligten Funktionen, aber das bedeutet noch lange nicht, dass man diese Ableitungen auch explizit vorliegen hat. Eine Berechnung dieser Ableitungen per Hand ist eine beliebte Fehlerquelle, insbesondere bei Optimierungsproblemen größerer oder auch nur mittlerer Dimension. Als mögliche Auswege bieten sich hier neben der Verwendung von Computeralgebra–Systemen wie MAPLE oder MATHEMATICA an:

- *numerische Differentiation*,
- *automatische Differentiation*.

Die numerische Differentiation mittels finiter Differenzen ist allerdings nicht ganz unproblematisch, siehe etwa [67, 45]. Für die Belange der Optimierung kommt der automatischen Differentiation daher in letzter Zeit immer größere Bedeutung zu; insbesondere ist die automatische Differentiation mittlerweile fester Bestandteil von *Modellierungssprachen* wie

GAMS [22] oder AMPL [58]. Der interessierte Leser findet einen Einstieg in das Thema der automatischen Differentiation in [154] und vor allem in [69], siehe auch [70] für geeignete Software.

(e) Dimension bzw. Besetztheitsstruktur des Problems:
Unter der Dimension eines Optimierungsproblems der Gestalt (1.4) versteht man die Größen von n, m und p, die bei der Auswahl eines geeigneten Verfahrens naturgemäß eine wichtige Rolle spielen. Ebenso bedeutend ist auch die Besetztheitsstruktur eines Optimierungsproblems. Beispielsweise hat man bei der Anwendung des Newton–Verfahren zur Lösung des unrestringierten Minimierungsproblems

$$\min f(x), \quad x \in \mathbb{R}^n,$$

in jedem Iterationsschritt ein lineares Gleichungssystem der Gestalt

$$\nabla^2 f(x^k) d = -\nabla f(x^k)$$

zu lösen, wobei x^k die aktuelle Iterierte des Verfahrens bezeichnet (vergleiche [66]). Dieses Gleichungssystem ist natürlich dann vergleichsweise einfach zu lösen, wenn die Dimension n sehr klein ist oder aber die Hesse–Matrix $\nabla^2 f(x^k)$ *schwach besetzt* (engl.: *sparse*) ist, d.h., sehr viele Nulleinträge besitzt (die möglichst noch an „schönen“ Stellen sein sollten).