

Optimale Steuerung des Klimas im Gewächshaus mit Evolutionären Algorithmen: Grundlagen, Verfahren und Ergebnisse

Hartmut Pohlheim

Technische Universität Ilmenau

Fakultät für Informatik und Automatisierung

Institut für Automatisierungs- und Systemtechnik

Fachgebiet Dynamik und Simulation ökologischer Systeme

PSF 327, D-98684 Ilmenau

Dr Adolf Heißner

Institut für Gemüse- und Zierpflanzenbau Großbeeren/Erfurt e.V.

Theodor-Echtermeyer-Weg 1, D-14979 Großbeeren

Es wird ein Gewächshausklimamodell zur Bestimmung von Temperatur, Luftfeuchte und Kohlendioxidkonzentration bei Anbau einer Gemüsepflanze beschrieben. Die Bilanzgleichungen für das Gewächshaus sowie die Gemüsepflanze werden dargestellt. Im weiteren wird auf die Ertragsoptimierung anhand dieses Modells eingegangen. Die notwendigen Grundlagen der Simulation und Optimierung werden erläutert. Für die Optimierung werden Evolutionäre Algorithmen verwendet. Alle Berechnungen wurden mit dem Softwarepaket `Matlab` durchgeführt.

Stichworte:

Gewächshaus, Klimamodell, Temperatur, Luftfeuchte, Kohlendioxidkonzentration, Gaswechsel, Transpiration, Tomate, Kurzzeitmodell, Wachstumsmodell, Bilanzen, Modellierung, Simulation, Optimierung, Evolutionäre Algorithmen, Genetic and Evolutionary Algorithm Toolbox, `Matlab`

Stand: 1996

Inhaltsverzeichnis

1 EINFÜHRUNG	3
2 MODELL DES GEWÄCHSHAUSKLIMAS	4
2.1 BILANZEN DES GEWÄCHSHAUSES	5
2.1.1 Parameter und Hilfsgrößen des Gewächshauses	5
2.1.2 Energiebilanz	5
2.1.3 Wasserdampfbilanz.....	6
2.1.4 Kohlendioxidbilanz.....	7
2.2 BILANZEN DES PFLANZENBESTANDES	8
2.2.1 Parameter des Pflanzenbestandes.....	8
2.2.2 Gaswechsel.....	8
2.2.3 Transpiration.....	9
2.3 ZIELGRÖßE	9
2.4 METEOROLOGISCHE DATEN	10
3 SIMULATION UND OPTIMIERUNG	11
3.1 SIMULATION.....	11
3.2 OPTIMIERUNG - EVOLUTIONÄRE ALGORITHMEN	11
3.2.1 Grundstruktur eines Evolutionären Algorithmus	12
3.2.2 Selektion	12
3.2.3 Rekombination.....	12
3.2.4 Mutation	13
3.2.5 Reinsertion	13
3.2.6 Migration.....	13
3.2.7 Verwendete Verfahren.....	14
4 ERGEBNISSE DER OPTIMIERUNG	15
4.1 OPTIMIERUNG IN 1-STUNDE-SCHRITTEN	15
4.1.1 Juni.....	15
4.1.2 April.....	17
4.1.3 Februar.....	18
5 AUSBLICK	19
6 ANHANG	20
6.1 LITERATUR.....	20
6.1.1 Wachstums- und Ertragsmodelle, Gewächshausklima.....	20
6.2.2 Simulation und Optimierung.....	21
6.2 TABELLEN DES GEWÄCHSHAUSMODELLES.....	23
6.3 TABELLEN DER BILANZGLEICHUNGEN	24
6.4 TABELLEN DES PFLANZENWACHSTUMSMODELLES.....	26
6.5 METEOROLOGISCHE DATEN	26
6.6 GRUNDLAGEN DER SIMULATION	28
6.6.1 Das EULERSche Polygonzugverfahren.....	28
6.6.2 Das klassische RUNGE-KUTTA-Verfahren	28
6.6.3 Eingebettete RUNGE-KUTTA-Verfahren.....	29
6.7 GRUNDLAGEN DER DYNAMISCHEN OPTIMIERUNG	32
6.7.1 Systembeschreibung	32
6.7.2 Beschränkungen	32
6.7.3 Gütekriterien	33
6.8 OPTIMIERUNGSVERFAHREN.....	33
6.8.1 Deterministische Suchverfahren.....	33
6.8.2 Deterministische Suchverfahren unter Verwendung des Gradienten.....	34

1 Einführung

Für die Optimierung der Einflußgrößen im Gewächshaus sind Wachstums- und Ertragsmodelle sowie Modelle des Gewächshausklimas zu einem unentbehrlichen Hilfsmittel geworden ([3], [4], [7], [8], [12], [16], [18], [21]). International haben die Bemühungen zur Entwicklung solcher Strategien zur weiteren Vervollkommnung der Gewächshausklimasteuerung geführt. Durch Simulation oder im pflanzenbaulichen Versuch konnten Vorteile gegenüber herkömmlichen Steuerungen mit festen Führungswerten nachgewiesen werden ([1], [2], [5], [11], [15], [19], [20]).

Die vorgeschlagenen Lösungen beziehen sich aber im allgemeinen nur auf einzelne Wachstumsfaktoren wie die Temperatur oder die Kohlendioxidkonzentration, was die Anwendungsmöglichkeiten einschränkt. In der Regel kann die Steuerung der verschiedenen technischen Einrichtungen nicht unabhängig voneinander vorgenommen werden. Mit der Lüftung beispielsweise werden Temperatur, Luftfeuchte und Kohlendioxidkonzentration gleichzeitig beeinflußt. Ein solches Verhalten ist in den zu entwickelnden Strategien entsprechend zu berücksichtigen.

Ausgehend von dieser Forderung wurde ein Modell entworfen, das die Vorhersage der Temperatur, der Luftfeuchte und der Kohlendioxidkonzentration in einem Zeitintervall von 15-60 Minuten gestattet. Es läßt sich damit für Kurzfriststeuerungsaufgaben wie Maximierung der Stoffproduktion oder Vermeidung von Streßzuständen einsetzen.

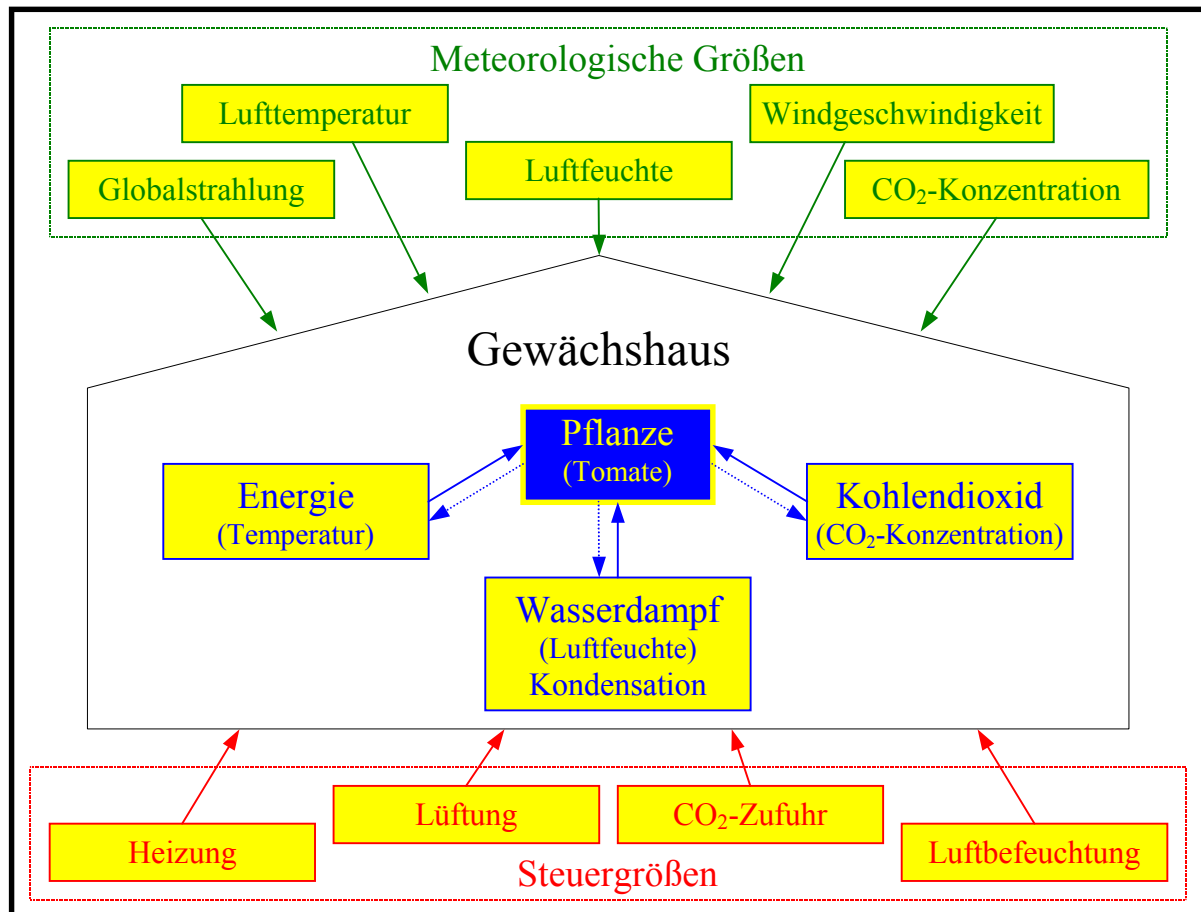
Die Anwendungsmöglichkeiten der gegenwärtigen Modellversion sollen an Hand von Simulations- und Optimierungsrechnungen dargestellt werden. Für die Simulation werden eingebettete Runge-Kutta-Verfahren verwendet. In der Optimierung kommen Evolutionäre Algorithmen zum Einsatz.

Im Abschnitt 2 wird das Gewächshausklimamodell zusammen mit dem Pflanzenwachstumsmodell (hier Tomate) beschrieben. Auf die Simulation und Optimierung wird im Abschnitt 3 eingegangen, speziell auf die Grundlagen der Optimierung mit Evolutionären Algorithmen. Die durchgeführten Optimierungen und ihre Ergebnisse werden im Abschnitt 4 dargestellt. In Abschnitt 5 wird eine Zusammenfassung sowie ein Ausblick auf weitere Entwicklungen gegeben. Abschnitt 6 listet die verwendete Literatur auf, enthält die Tabellen mit den genauen Größen und Gleichungen aus Abschnitt 2 sowie die Grundlagen der Simulation und Optimierung dynamischer Systeme aus Abschnitt 3.

2 Modell des Gewächshausklimas

Das Gewächshausklimamodell beschreibt die Abhängigkeit der Temperatur, der Luftfeuchte und der Kohlendioxidkonzentration im Gewächshaus von den Stellgrößen sowie den mikrometeorologischen Bedingungen im Freien mit einem System von 3 nichtlinearen Differentialgleichungen 1. Ordnung.

Fig. 1: Schema des Gewächshausklimamodells



Die 3 Differentialgleichungen des Modells sind die Bilanzgleichungen im Gewächshausinnenraum für

- Energie (Temperatur), Equ. 1,
- Wasserdampf (Luftfeuchte), Equ. 2 und
- Kohlendioxid (Kohlendioxidkonzentration), Equ. 6.

Als Stellgrößen zur Beeinflussung des Gewächshauses kommen zur Anwendung:

- Heizung, Q [W/m^2]
- Lüftung, LR [m^3/m^2],
- Kohlendioxidzufuhr, W [$\text{g}/(\text{m}^2 \cdot \text{h})$] und
- Luftbefeuchtung, RM [$\text{g}/(\text{m}^2 \cdot \text{h})$].

Die zu messenden mikrometeorologischen Bedingungen im Freien sind:

- Globalstrahlung, $IGLOB$ [W/m^2],
- Lufttemperatur außen, $TEMA$ [$^{\circ}\text{C}$],
- Luftfeuchte außen, FA [% r.F.],

- Kohlendioxidkonzentration außen, CA [ppm] und
- Windgeschwindigkeit, U [m/s].

Da in diesem Modell nur ein Kompartiment betrachtet wird (Gewächshausinnenraum mit Pflanzenbestand), müssen neben den Bedingungen im Freien weitere Eingangsgrößen berücksichtigt werden. Dies sind

- Temperatur der Bodenoberfläche, TEMB [°C], (Tab. 7) und
- Temperatur der Gewächshaushülle, TEMG [°C], (Tab. 7).

Beide Größen sind genauso wie die meteorologischen Bedingungen im Freien vorzugeben bzw. zu messen. Eine weitere Möglichkeit ist die Darstellung durch ein Regressionsmodell, das die Abhängigkeit von den meteorologischen Bedingungen wiedergibt.

Die in den Bilanzgleichungen verwendete Temperaturgröße ist durch die spezifische Wärmekapazität von Luft und Pflanzen (Volumen pro Grundfläche und Wasseräquivalent des Pflanzenbestandes) im Gewächshaus definiert und stellt damit ein Mittel aus Luft- und Pflanzentemperatur dar.

Für Berechnungen mit dem Modell sind neben einer Reihe von physikalischen Konstanten (Tab. 4) Gewächshausparameter (Tab. 6) und -hilfsgrößen (Tab. 7) sowie Koeffizienten für die Darstellung der Transpiration (Tab. 2, [13]) und des Kohlendioxidgaswechsels (Tab. 1, [14]) durch Modelle und Pflanzenbestandsparameter (Tab. 11) in die Bilanzgleichungen einzusetzen.

2.1 Bilanzen des Gewächshauses

2.1.1 Parameter und Hilfsgrößen des Gewächshauses

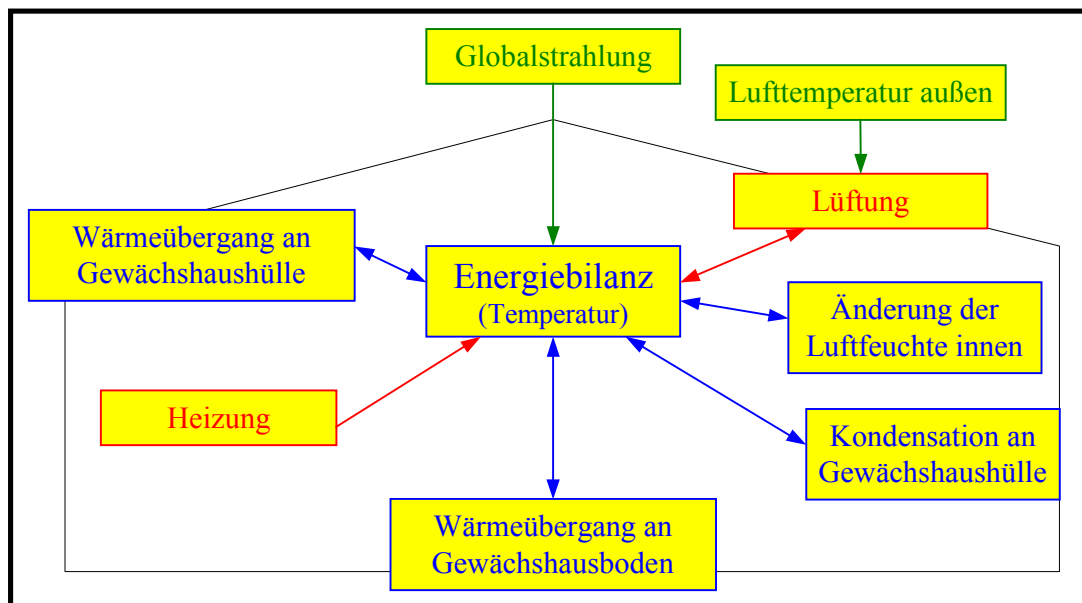
Das Gewächshaus ist durch eine Reihe von Parametern (Tab. 6) und Hilfsgrößen (Tab. 7) charakterisiert.

2.1.2 Energiebilanz

Die Energiebilanz wird durch folgende Komponenten dargestellt (Tab. 8):

1. Wärmezufuhr durch Heizen
2. Erwärmung durch Globalstrahlung
3. Änderung der Luftfeuchte
4. Energieaustausch mit der Umgebung durch Lüftung
5. Wärmedurchgang durch die Gewächshaushülle
6. Wärmeübertragung an der Gewächshausbodenoberfläche
7. Kondensation/Verdunstung von Wasser an der Gewächshaushülle
8. Wärmekapazität des Gewächshauses (Pflanzenbestand, Luft)

Fig. 2: Schema der Energiebilanz



Die Änderung der Lufttemperatur (Energiebilanz) berechnet sich mit den aufgeführten Größen nach folgender Gleichung.

Equ. 1: Energiebilanz, DTEMI, [K/s]

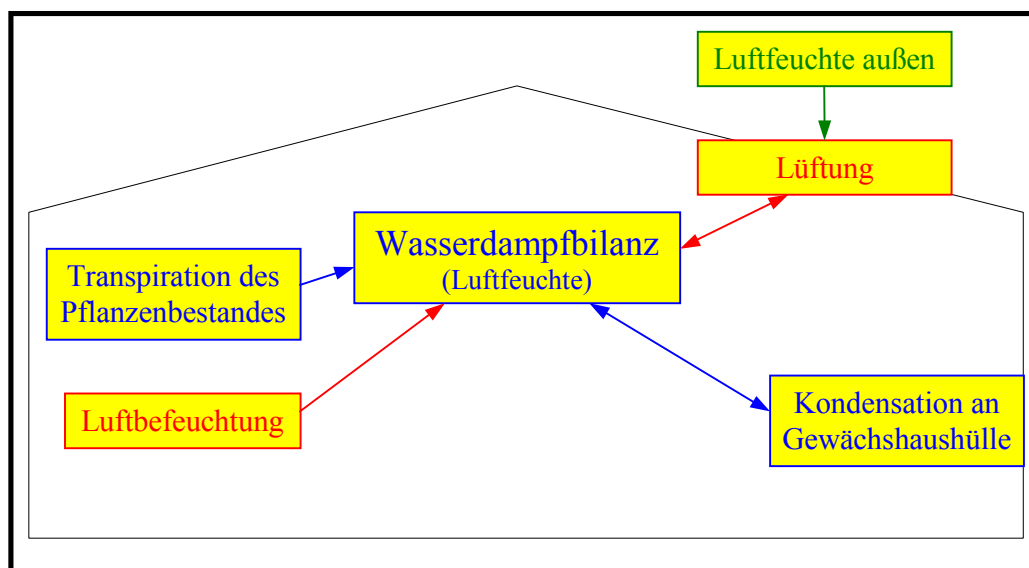
$$DTEMI = (1 / (CG + GH \cdot CPD \cdot DDI)) \cdot (Q + q_{glob} - q_{luwe} - q_{duga} - q_{boden} - q_{konden} - GH \cdot (VDW0 + CPD \cdot TEMI) \cdot DDDI)$$

2.1.3 Wasserdampfbilanz

Die Wasserdampfbilanz wird durch folgende Komponenten dargestellt (Tab. 9):

1. Transpiration des Pflanzenbestandes
2. Luftbefeuchtung
3. Wasserdampfaustausch mit der Umgebung durch Lüftung
4. Kondensation/Verdunstung an der Gewächshaushülle

Fig. 3: Schema der Wasserdampfbilanz



Die Änderung der Wasserdampfdichte (Wasserdampfbilanz) berechnet sich mit den aufgeführten Größen nach folgender Gleichung.

Equ. 2: Wasserdampfbilanz, DDDI, $[\text{g}/(\text{m}^3 \cdot \text{s})]$

$$\text{DDDI} = (1 / (\text{GH} \cdot 3600)) \cdot (\text{trans} + \text{lube} - \text{wadawe} - \text{kondverd})$$

Kondensatbilanz

Kondensation (Verdunstung) an der Gewächshülle tritt auf, wenn der Funktionswert von kondens (Tab. 9) positiv (negativ) ist. Kondensation ist immer möglich, bei Überschreiten der maximalen Kondensatflächendichte (KS) fließt Kondensat an der Gewächshaushülle ab. Verdunstung kann dagegen nur auftreten, wenn Kondensat an der Gewächshaushülle vorhanden ist.

Equ. 3: Kondensation ($\text{konden} > 0$):

$$\text{kondverd} = \text{konden},$$

Equ. 4: Verdunstung ($\text{konden} < 0$):

$$\text{wenn KS} > 0: \text{kondverd} = \text{konden},$$

$$\text{wenn KS} = 0: \text{kondverd} = 0.$$

Equ. 5: Berechnung der Kondensatflächendichte, KS, $[\text{g}/\text{m}^2]$:

$$\text{KS} = \text{KSA} + \int \text{kondverd} \, dt, \text{ (KSA: Anfangswert),}$$

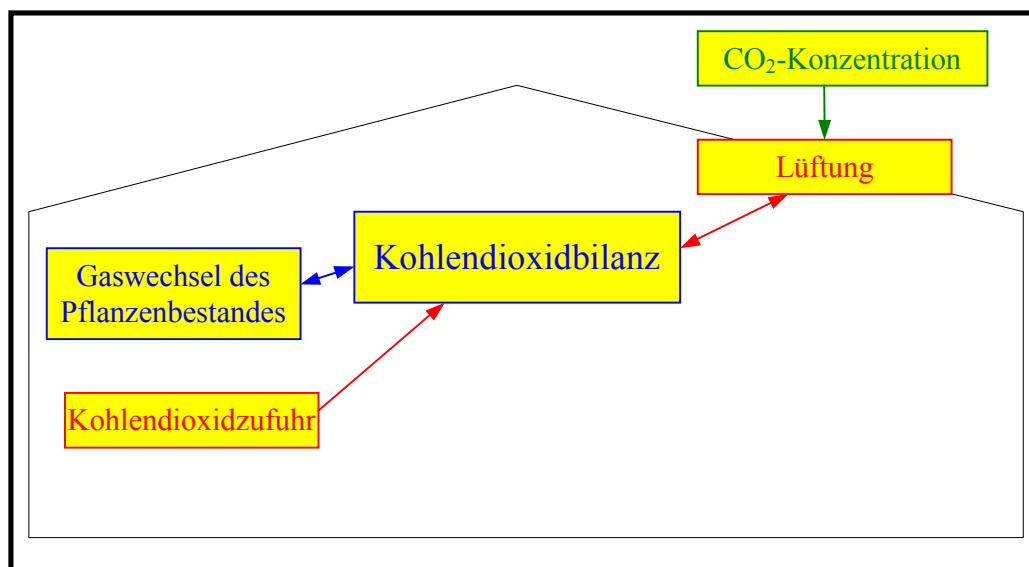
$$\text{KS}_{\text{Minimum}} = 0, \text{KS}_{\text{Maximum}} = \text{KSM} \text{ (KSM: maximale Kondensatflächendichte).}$$

2.1.4 Kohlendioxidbilanz

Die Kohlendioxidbilanz wird durch folgende Komponenten dargestellt (Tab. 10):

1. Kohlendioxidzufuhr durch Begasung
2. Kohlendioxidgaswechsel (Photosynthese und Respiration) des Pflanzenbestandes
3. Kohlendioxidaustausch mit der Umgebung durch Lüftung

Fig. 4: Schema der Kohlendioxidbilanz



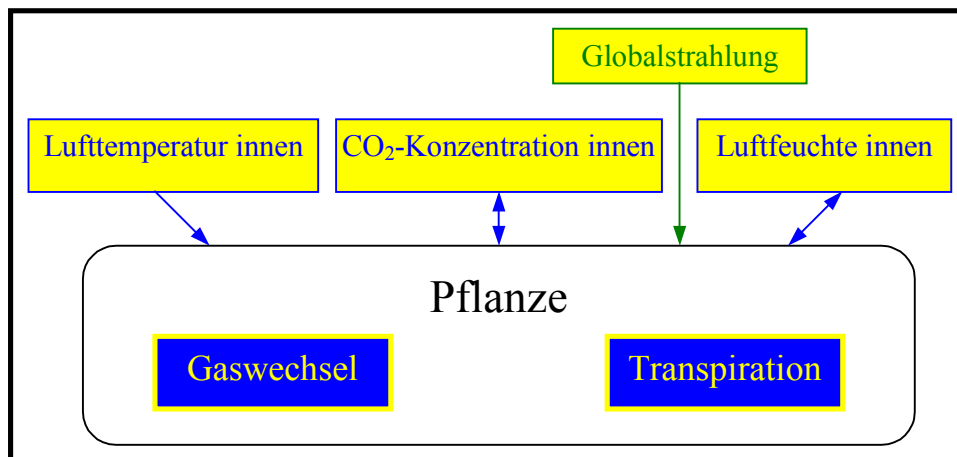
Die Änderung der Kohlendioxidkonzentration (Kohlendioxidbilanz) berechnet sich mit den aufgeführten Größen nach folgender Gleichung.

Equ. 6: Kohlendioxidbilanz, DCI, $[\text{ppm}/\text{s}]$

$$\text{DCI} = (1 / (10^{-6} \cdot \text{DC} \cdot \text{GH} \cdot 3600)) \cdot (\text{W-gawe} - \text{kodiwe})$$

2.2 Bilanzen des Pflanzenbestandes

Fig. 5: Schema des Pflanzenwachstumsmodelles



2.2.1 Parameter des Pflanzenbestandes

Der Pflanzenbestand ist durch eine Reihe von Parametern (Tab. 11) charakterisiert.

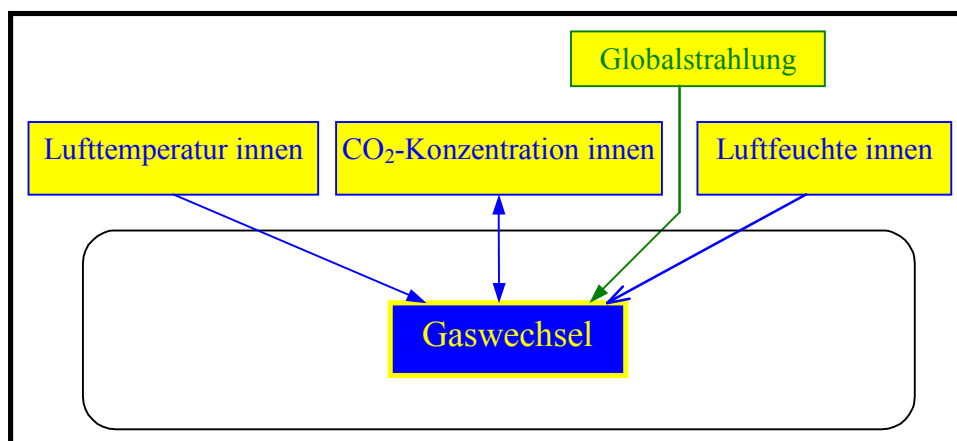
2.2.2 Gaswechsel

Für die Berechnung des Gaswechsels der Blattfläche des Pflanzenbestandes wird folgendes Modell verwendet.

Equ. 7: Gaswechsels der Blattfläche, g_{awebfl} , [mg/(dm²·h)]

$$g_{awebfl} = P_0 \cdot FICT \cdot FSD$$

Fig. 6: Schema des Gaswechsels



Die Hilfsgröße FICT wird mittels einer parametrischer Gleichung berechnet.

$$FICT = C_1 \cdot (1 - \exp(-C_2 \cdot 0.5 \cdot I)) \cdot (1 - \exp(-C_3 \cdot CI)) \cdot (TEMI + C_4 \cdot TEMI^2) - C_5 \cdot (TEMI + C_6 \cdot TEMI^2);$$

Tab. 1: Konstanten des Gaswechsel

C ₁	C ₂	C ₃	C ₄	C ₅	C ₆	SDIG ₁	SDIG ₂	EE ₁	EE ₂
0.1381	8.687·10 ⁻³	3.697·10 ⁻³	-1.908·10 ⁻²	2.073·10 ⁻³	8.752·10 ⁻²	5 hPa	10 hPa	10 ⁻⁴	10 ⁻³

Für die Hilfsgröße FSD werden 3 Bereiche in Abhängigkeit von SDI unterschieden.

$SDI < SDIG_1$	$FSD = \exp(EE_1 \cdot (SDIG_1 - SDI)^2)$
$SDIG_1 \leq SDI \leq SDIG_2$	$FSD = 1$
$SDI > SDIG_2$	$FSD = \exp(EE_2 \cdot (SDIG_2 - SDI)^2)$

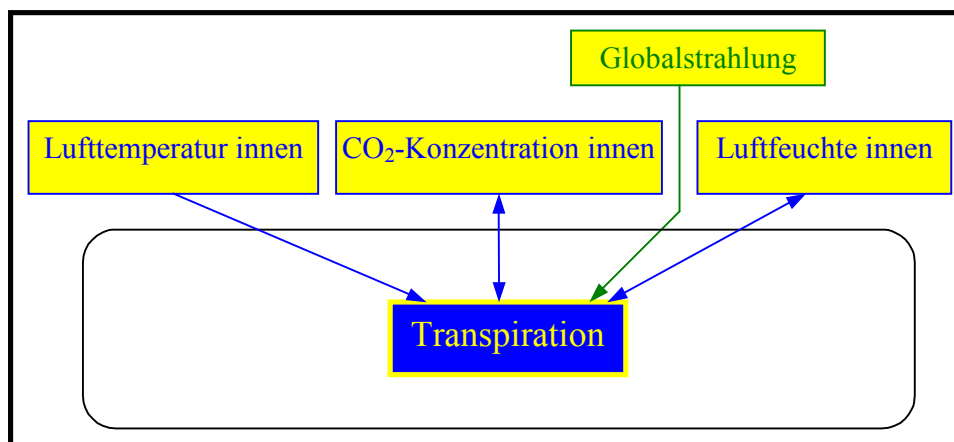
2.2.3 Transpiration

Für die Berechnung der Transpiration der Blattfläche des Pflanzenbestandes wird folgendes Modell verwendet.

Equ. 8: Transpiration der Blattfläche, tr_{blfl} , [g/(dm²·h)]

$$tr_{blfl} = V_0 \cdot VREL;$$

Fig. 7: Schema der Transpiration



Die Hilfsgrößen werden mittels parametrischer Gleichungen berechnet.

$$VREL = VRELC \cdot SDI \cdot (B_1 + B_2 \cdot I + B_3 \cdot I^2 + B_4 \cdot (PDI/PSI) \cdot 100) / VSTAN,$$

$$VRELC = 1 - B_0 \cdot (CI - 600),$$

$$VSTAN = 10 \cdot (B_1 + B_2 \cdot 300 + B_3 \cdot 300^2 + B_4 \cdot 60).$$

Tab. 2: Konstanten der Transpiration

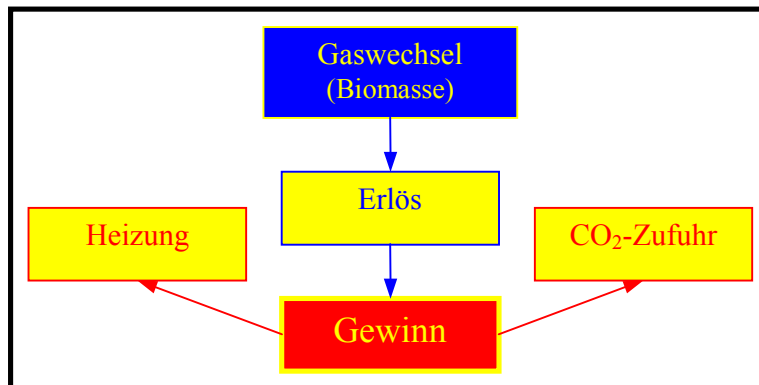
B_0	B_1	B_2	B_3	B_4
$5.400 \cdot 10^{-4}$	$-2.219 \cdot 10^{-4}$	$5.213 \cdot 10^{-6}$	$-6.623 \cdot 10^{-9}$	$8.500 \cdot 10^{-6}$

2.3 Zielgröße

Als Zielgröße ist der Gewinn definiert. Dieser wird aus folgenden Größen gebildet:

1. Erlös (Verkaufspreis), berechnet aus der Trockensubstanzproduktion und
2. Kosten für Heizung und Kohlendioxidbegasung.

Fig. 8: Schema der Zielgröße



Tab. 3: Komponenten der Zielgröße

Zielgröße	Symbol	Einheit	Funktion
Trockensubstanzbildung	BIOM	g/(m ² ·h)	gawe · (30 / 44)
Gemüsepreis	PR1	DM/kg	3
CO ₂ -Preis	PR2	DM/kg	1
Energiepreis	PR3	DM/(W·h)	0.00005

BIOM ist die aus dem Kohlendioxidgaswechsel berechnete Trockensubstanzproduktion, die in eine Ertragsgröße und nachfolgend in eine Erlösgröße umgerechnet wird. Die Kosten für die Kohlendioxidzufuhr werden aus dem Kohlendioxidpreis errechnet. Analog erfolgt die Berechnung der Wärmezufuhr durch Heizen über den Energiepreis für Heizöl.

Equ. 9: Gewinnänderung, GEWI, [0.01 DM / (m²·h)]:

$$GEWI = (BIOM \cdot 0.5 \cdot (1 / 0.05) \cdot 10^{-3} \cdot PR1 - W \cdot PR2 / 1000 - Q \cdot PR3) \cdot 100$$

2.4 Meteorologische Daten

Den Berechnungen liegen Klimadaten für Temperatur (Figure 19), Luftfeuchte (Figure 20), Globalstrahlung (Figure 21) und Windgeschwindigkeit (Figure 22) zugrunde. Für die Monate Februar, April und Juni wurde je ein Tag als repräsentatives Beispiel verwendet. Die Daten sind langjährige Mittel der jeweiligen Monate (Potsdam; 1951-1970).

3 Simulation und Optimierung

3.1 Simulation

Das Gewächshausklimamodell ist als System von Differentialgleichungen (DGL) gegeben. Die Simulation entspricht der Lösung dieses Differentialgleichungssystems (DGLS) über den interessierenden Zeitraum. Oft wird dies als Lösung eines Anfangswertproblems bezeichnet.

Mit den bekannten Werten der Variablen des DGLS zum Beginn des Zeitraums hat das DGLS eine exakte Lösung zu jedem späteren Zeitraum. Allerdings läßt sich diese exakte Lösung nur in den seltensten Fällen direkt ermitteln. Numerische Algorithmen (Integrationsverfahren) ermöglichen eine schnelle und genaue Lösung solcher Probleme.

Für eine Einführung in die Grundlagen dieser Verfahren sei auf den Anhang verwiesen. Dort werden verschiedene Verfahren dargestellt und ausführlich erläutert.

Für die Integration des vorliegenden Modells wurde ein eingebettetes Runge-Kutta-Verfahren 4.(5.) Ordnung verwendet.

3.2 Optimierung - Evolutionäre Algorithmen

Für das oben dargestellte Modell soll der Verlauf einer Steuerung unter Beachtung der Zustandsgleichungen ermittelt werden. Damit kommt die dynamische Optimierung zur Anwendung, d.h., es wird eine optimale Steuerung in einem bestimmten Zeitintervall gesucht.

Für eine Einführung in die Grundlagen der Optimierung dynamischer Systeme sei auf den Anhang verwiesen. Dort wird eine mathematische Beschreibung dynamischer Systeme zusammen mit Beschränkungen und Gütekriterien gegeben. Danach folgt eine kurze Erläuterung der Funktionsweise von deterministischen Suchverfahren ohne/mit Verwendung von Gradienteninformationen.

In diesem Abschnitt wird eine Einführung in die Verfahren der Evolutionären Algorithmen gegeben, die in den letzten Jahren als Alternative zu den deterministischen Suchverfahren verstärkt Anwendung fanden.

Evolutionäre Algorithmen unterscheiden sich von den deterministischen Suchverfahren vor allem in der Art der Veränderung der Parameter/Variablen. Evolutionäre Algorithmen sind stochastische Suchverfahren, die mit Prinzipien der natürlichen Evolution arbeiten. Sie basieren auf zufällig verteilten Regeln und modellieren natürliche Prozesse, z.B.:

- Populationen
- Selektion
- Rekombination
- Mutation
- Migration

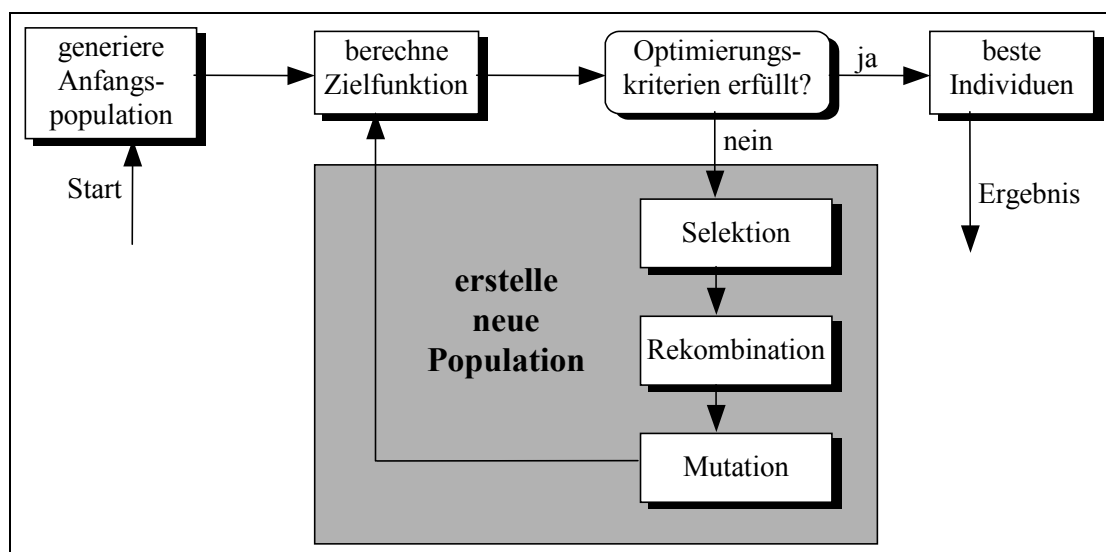
Durch die Verwendung von Populationen von Individuen anstelle einzelner Lösungen wird die Suche in einer parallelen Art und Weise durchgeführt.

3.2.1 Grundstruktur eines Evolutionären Algorithmus

Zu Beginn der Berechnungen wird eine Anzahl von Individuen (die Population) zufällig initialisiert (jeder Variablen wird ein zufälliger Wert aus dem möglichen Intervall zugewiesen). Dann wird die Zielfunktion für diese Individuen berechnet. Damit ist die erste Generation produziert.

Individuen werden entsprechend ihrer Fitneß für die Produktion von Nachkommen ausgewählt. Immer zwei Eltern rekombinieren zur Produktion von zwei Nachkommen. Alle Nachkommen werden mit einer vorgegebenen Wahrscheinlichkeit mutiert und nachfolgend wird deren Fitneß berechnet. Die Nachkommen werden in die Population eingefügt und ersetzen die Eltern, womit eine neue Generation produziert ist. Diese Schleife wird solange durchlaufen, bis ein Abbruchkriterium erreicht ist. Das beste gefundene Individuum ist die Lösung, das gefundene Optimum.

Fig. 9: Grundstruktur eines Evolutionären Algorithmus



Der Evolutionäre Algorithmus für eine Population ist leistungsfähig und erreicht für eine große Klasse von Problemen gute Ergebnisse.

Im folgenden wird ein Überblick über die einzelnen Schritte eines Evolutionären Algorithmus gegeben.

3.2.2 Selektion

In der Selektion werden die Individuen (Eltern) für die Fortpflanzung (Rekombination) ausgewählt und die Anzahl der jeweils zu produzierenden Nachkommen festgelegt. Der erste Schritt ist die Fitneßzuweisung (proportional fitness assignmen, rank-based fitness assignment). Die direkte Selektion erfolgt im nächsten Schritt. Eltern werden entsprechend ihrer Fitneß durch einen Selektionsalgorithmus (roulette-wheel selection, stochastic universal sampling, lokale Selektion, truncation selection, tournament selection) ausgewählt.

3.2.3 Rekombination

Durch Rekombination werden neue Individuen produziert, die Teile der Information von den Eltern in sich vereinen. Rekombination findet im allgemeinen zwischen Paaren von Individuen (den selektierten Eltern) statt. Jedes Paar produziert pro Generation zwei

Nachkommen. Je nach Art der Variablen gibt es verschiedene Verfahren. Ein einfaches aber dennoch leistungsfähiges Verfahren ist die diskrete Rekombination.

Diskrete Rekombination

führt einen Austausch der Variablenwerte zwischen den Individuen durch. Für jede Variable eines Nachkommen wird zufällig und mit gleicher Wahrscheinlichkeit entschieden, welcher Elter seinen Variablenwert an diesen Nachkommen weitergibt. Diskrete Rekombination kann die Ecken eines Hyperwürfels generieren, der durch die Variablenwerte der Eltern definiert ist. Diskrete Rekombination kann für jede Art von Variablen genutzt werden, egal ob in reeller oder binärer Repräsentation.

Weitere Verfahren sind z.B.

- intermediate recombination [35],
- line recombination [35],
- extended line recombination [22],
- multi point crossover [22],
- shuffle crossover [22].

3.2.4 Mutation

Nach der Rekombination wird jeder Nachkomme mutiert. Die Variablen der Nachkommen werden durch die Addition kleiner Zufallswerte (die Mutationsschrittweite) zu den Variablen mutiert. Dieser Prozeß findet mit einer niedrigen Wahrscheinlichkeit statt, wobei die Wahrscheinlichkeit für das Mutieren einer Variablen mit zunehmender Variablenzahl pro Individuum abnimmt, die Mutationswahrscheinlichkeit ist invers proportional zur Variablenzahl.

3.2.5 Reinsertion

Wenn die Nachkommen durch Selektion, Rekombination und Mutation aus den Individuen der alten Population produziert sind, wird die Fitneß der Nachkommen bestimmt. Durch ein Reinsertion-Verfahren wird entschieden, welche Nachkommen in die Population eingefügt werden und welche Individuen der alten Population ersetzt werden.

3.2.6 Migration

In der natürlichen Evolution tendieren Arten dazu, sich in Unterpopulationen der Gesamtpopulation fortzupflanzen, die bis zu einem gewissen Grad voneinander isoliert sind. Fortpflanzung zwischen Individuen verschiedener Unterpopulationen ist möglich, wahrscheinlicher ist aber, daß sich die Individuen innerhalb einer Unterpopulation fortpflanzen.

Im Migrationsmodell wird die Gesamtpopulation in mehrere Unterpopulationen eingeteilt, die sich für eine festzulegende Zeit oder Anzahl von Generationen (Isolationszeit) unabhängig voneinander entwickeln. Nach dieser Zeit werden einige Individuen zwischen den Unterpopulationen ausgetauscht (Migration). Die Anzahl der auszutauschenden Individuen (die Migrationsrate), die Art der Auswahl der Individuen für die Migration und das Schema der Migration entscheiden über die genetische Vielfalt, die sich in den Unterpopulationen entwickeln kann und den Austausch an Informationen zwischen den Unterpopulationen.

Die Migration der Individuen zwischen den Unterpopulationen kann stattfinden:

- zwischen allen Unterpopulationen (komplette Netzstruktur - uneingeschränkt),
- in einer Nachbarschaftsstruktur,
- in einer Ringstruktur.

Die Unterteilung in Unterpopulationen und die Migration verbessert die Ergebnisse gegenüber des Algorithmus mit nur einer Population.

3.2.7 Verwendete Verfahren

Für die Optimierung des Gewächshausklimamodells wurden Evolutionäre Algorithmen genutzt. Diese bieten den Vorteil, daß außer dem Modell für die Simulation des Systems keine weiteren problemspezifischen Informationen benötigt werden. Zudem sind Evolutionäre Algorithmen robust.

Die gesamten Berechnungen wurden unter Matlab durchgeführt. Die *Genetic and Evolutionary Algorithm Toolbox for use with Matlab* ([22], [37]) stand zur Verfügung.

Die Optimierung wurde mit dem *Multi Population Genetic Algorithm* [37] durchgeführt. Dabei wurden die folgenden evolutionären Operatoren und Parameter verwendet:

- linear ranking (selective pressure: 1.3)
- Selektion mit stochastic universal sampling oder truncation selection
- generation gap: 0.9
- discrete recombination
- mutation of the breeder genetic algorithm
- fitness-based reinsertion
- migration (ring structure; migration rate: 0.2; migration time: alle 20 Generationen)

Die Anzahl der Unterpopulationen und Individuen pro Unterpopulation wurden in Abhängigkeit der Größe des Optimierungsproblems gesetzt. Diese Werte werden bei der Beschreibung der jeweiligen Optimierung mit genannt.

4 Ergebnisse der Optimierung

4.1 Optimierung in 1-Stunde-Schritten

Die erste durchgeführte Optimierung wurde mit einer für jeweils eine Stunde konstanten Steuerung durchgeführt. Um die Dimension noch kleiner zu halten, wurde jeweils nur eine Stunde optimiert. Nach Beendigung der Optimierung einer Stunde wurden die erreichten Zustandswerte als Startwerte der nächsten Stunde verwendet. Damit war die Dimension des Optimierungsproblems 3 und somit ohne Probleme durchführbar. Eine Optimierung mit 150 Individuen, 1 Population und 40 Generationen war mehr als ausreichend.

Diese Rechnung sollte mehr als Einstieg in die Optimierungsrechnungen dienen. Mit den hier gewonnenen Erfahrungen soll an weitere Rechnungen gegangen werden.

Zur Auswertung werden in allen folgenden Rechnungen die Verläufe der im Gewächshaus gemessenen Parameter (Dampfdichte DDI, Temperatur TEMI, Kondensat KOND; Kohlendioxidgehalt CI, Feuchte FI), die Steuerwerte (Heizung Q, Lüftung LR, Kohlendioxidbegasung W) und die Zielfunktion (Biomasse BIOM, Gewinn GEWI) dargestellt.

4.1.1 Juni

Fig. 10: Tagesgang von Dampfdichte, Temperatur und Kondensat innen für Juni

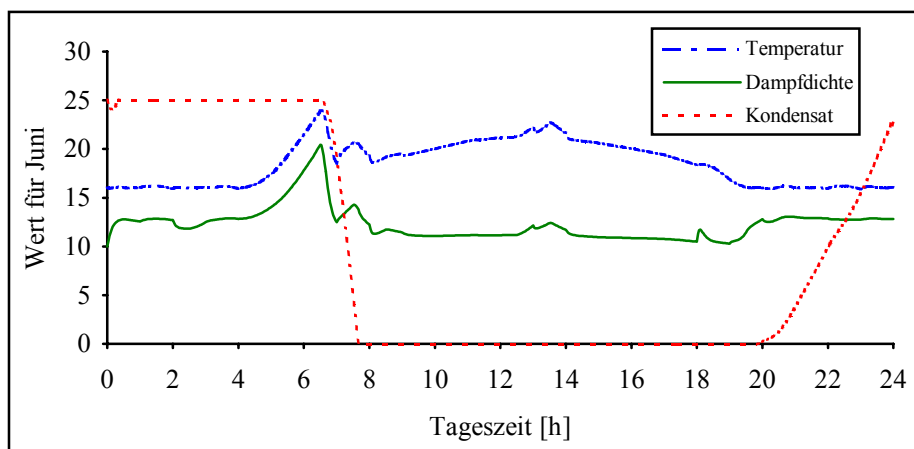


Fig. 11: Tagesgang von Kohlendioxidgehalt und Feuchte innen für Juni

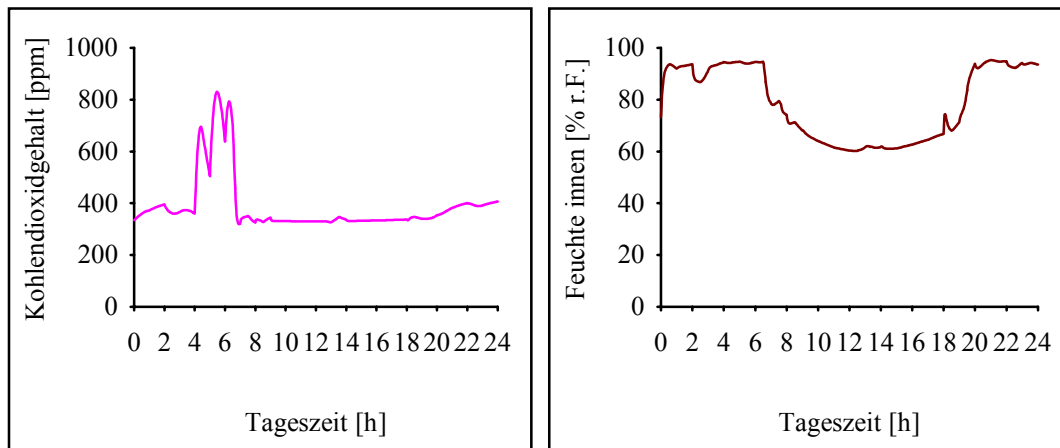


Fig. 12: Tagesgang von Heizung, Lüftung und Kohlendioxidzufuhr für Juni

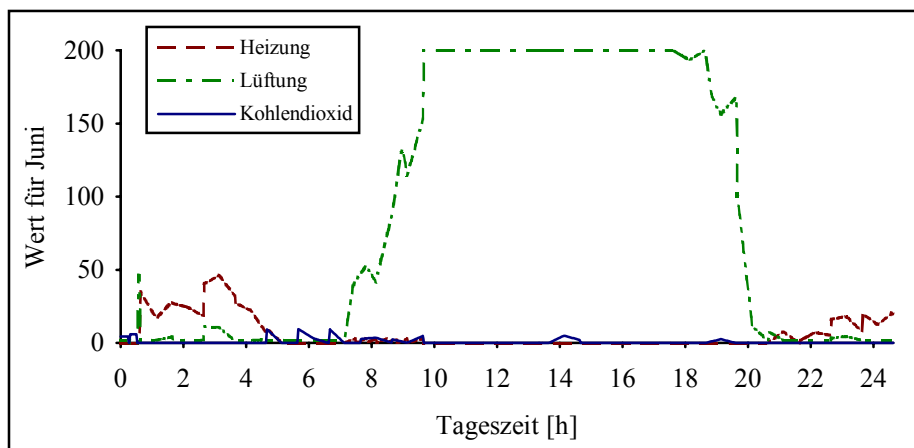
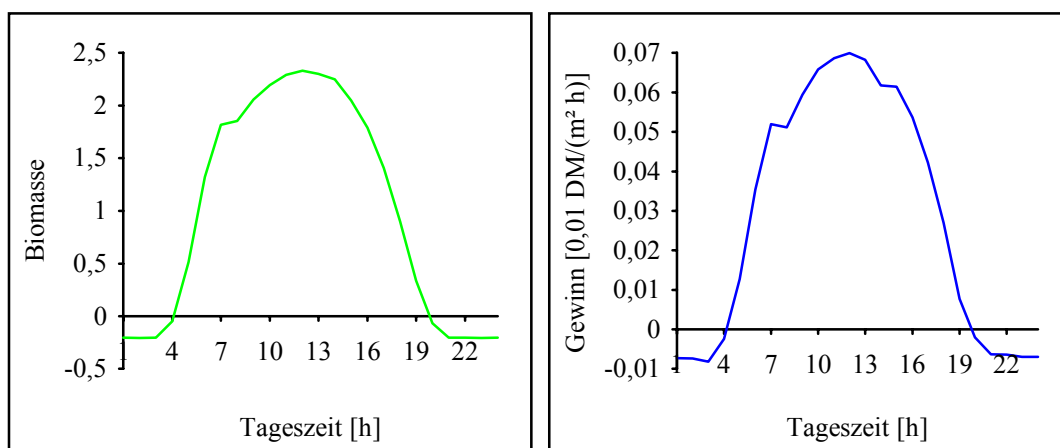


Fig. 13: Tagesgang von Biomasseprodukt und Gewinn pro Stunde für Juni



4.1.2 April

Fig. 14: Tagesgang von Dampfdichte, Temperatur und Kondensat innen für April

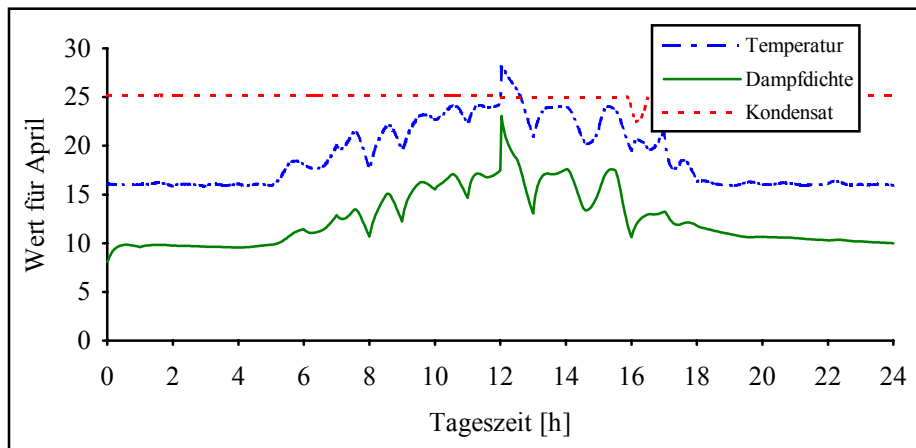


Fig. 15: Tagesgang von Kohlendioxidgehalt und Feuchte innen für April

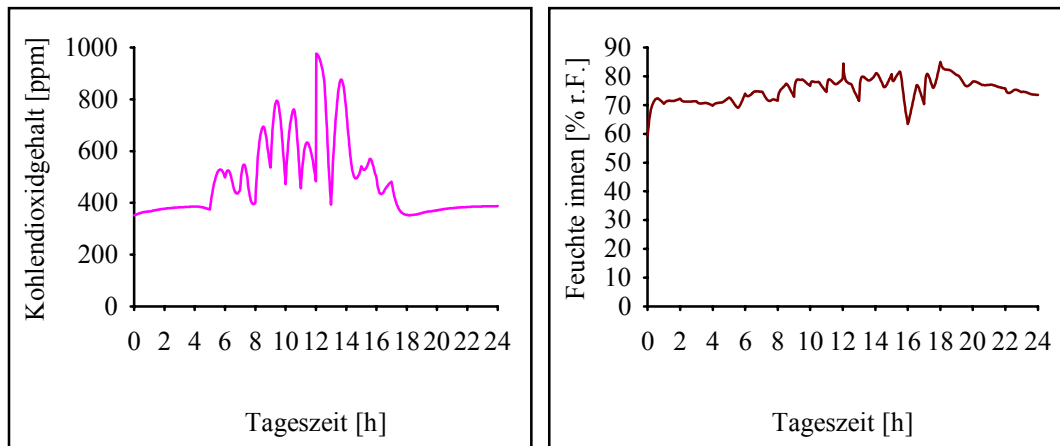


Fig. 16: Tagesgang von Heizung, Lüftung und Kohlendioxidzufuhr für April

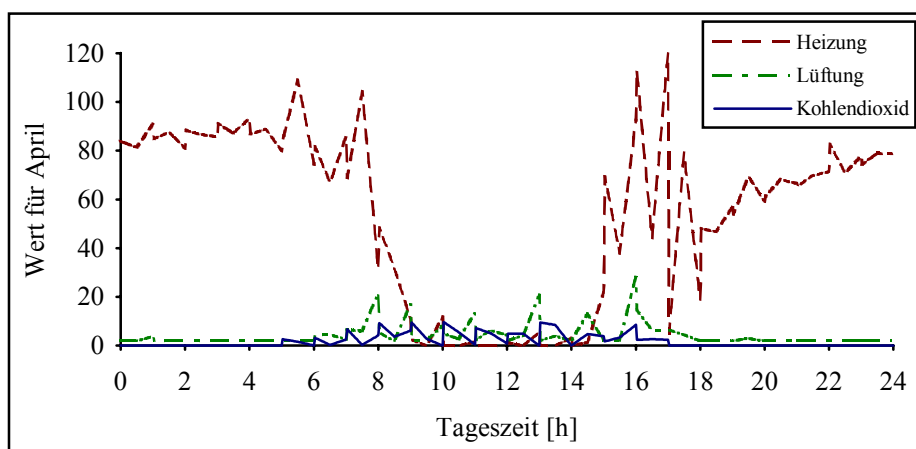
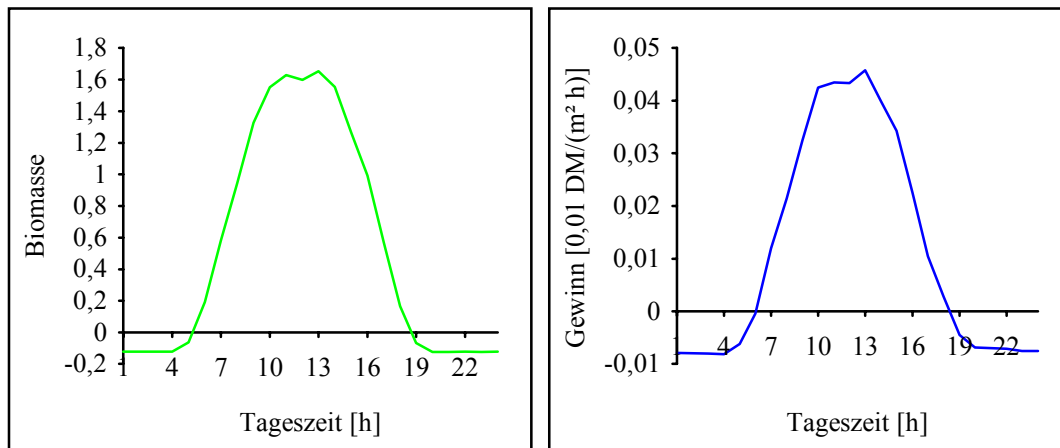
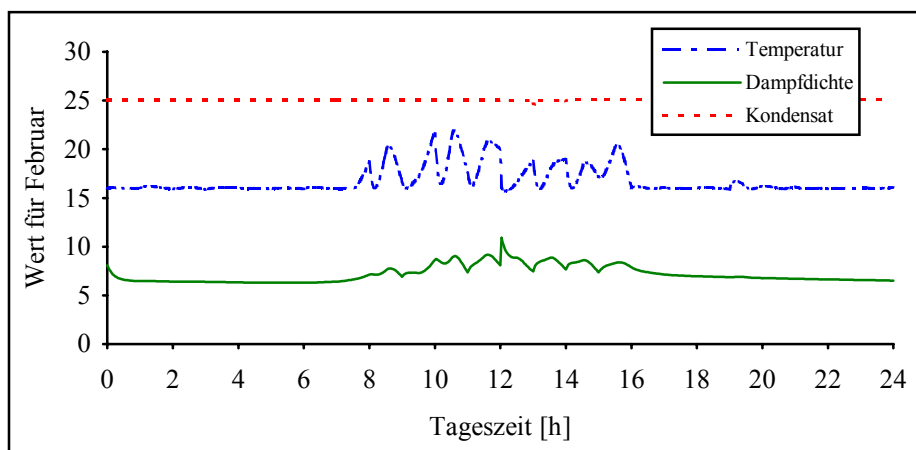


Fig. 17: Tagesgang von Biomasseprodukt und Gewinn pro Stunde für April



4.1.3 Februar

Fig. 18: Tagesgang von Dampfdichte, Temperatur und Kondensat innen für Februar



5 Ausblick

6 Anhang

6.1 Literatur

6.1.1 Wachstums- und Ertragsmodelle, Gewächshausklima

- [1] *Aikman, D. P. und Picken, A. J. F.*: Wind-related temperature setting in glasshouses. *Journal of Horticultural Science* 64, pp. 649-654, 1989.
- [2] *Bailey, B. J. und Seginer, I.*: Optimum control of greenhouse heating. *Acta Horticulturae*, 254, pp. 512-518, 1989.
- [3] *Bailey, B. J.*: Greenhouse models and climate control. Proceedings of the international seminar and British-Israel workshop on greenhouse technology, Bet Dagan, Israel, 26 March-2 April 1990, 1990.
- [4] *Bailey, B.*: Climate modelling and control in greenhouses. 173-201, 1991.
- [5] *Bakker, J. C.*: A CO₂ control algorithm based on simulated photosynthesis and ventilation rate. *Acta Horticulturae*, 174, pp. 387-392, 1985.
- [6] *Bot, G. P. A.*: A validated physical model of greenhouse climate. *Acta Horticulturae*, 245, pp. 389-396, 1989.
- [7] *Bot, G. P. A.*: Physical modelling of greenhouse climate. The computerized greenhouse, Automatic control application in plant production, pp. 51-73, 1993.
- [8] *Challa, H. und van Straten, G.*: Optimal diurnal climate control in greenhouses as related to greenhouse management and crop requirements. The computerized greenhouse, Automatic control application in plant production, pp. 119-137, 1993.
- [9] *Cierpinski, W., Broneske, J., Krüger, I., Kleisinger, S., Herbold, J. und Reisinger, G.*: Entwicklung eines Gewächshausystems mit minimiertem Energiebedarf und minimierter Umweltbelastung. Forschungsabschlußbericht zur Aufgabe BMFT-FB 0329192A, Institut für Gemüse- und Zierpflanzenbau Großbeeren/Erfurt e.V., 1994.
- [10] *Ehler, N.*: An autocalibrating model for simulating and measuring net canopy photosynthesis using standard greenhouse climate computer. *Computers and Electronics in Agriculture*, 6, pp. 1-20, 1991.
- [11] *Ehler, N. und Karlsen, P.*: OPTICO - a model based real-time expert system for dynamic optimization of CO₂ enrichment of greenhouse vegetable crops. *Journal of Horticultural Science*, 68, pp. 485-494, 1993.
- [12] *Gauthier, L.*: An object-oriented greenhouse simulation software system - Architecture and functional description - Description, validation and use of a simple model. *Canadian Agricultural Engineering*, 35, pp. 223-228, 1993.
- [13] *Heißner, A.*: Die Transpiration von Tomatenpflanzen im Gewächshaus in Abhängigkeit vom Dampfdrucksättigungsdefizit der Luft und der Bestrahlungsstärke. Versuchsbericht 93 3 23, Institut für Gemüse- und Zierpflanzenbau Großbeeren/Erfurt e.V., 1993.
- [14] *Heißner, A.*: Gewächshausklimamodell. Programmdokumentation, Institut für Gemüse- und Zierpflanzenbau Großbeeren/Erfurt e.V., 1994.

- [15] *Houter, G.*: Simulation of CO₂ consumption, heat demand and crop production of greenhouse tomato at different CO₂ strategies. *Acta Horticulturae*, 268, pp. 157-164, 1990.
- [16] *Jolliet, O.*: HORTITRANS, a model for predicting and optimizing humidity and transpiration in the greenhouses. *Journal of Agricultural Engineering Research*, 57, pp. 23-37, 1994.
- [17] *Jones, J. W., Hwang, Y. K. and Seginer, I.*: Simulation of Greenhouse Crops, Environments and Control Systems. *Acta Horticulturae*, 399, pp. 73-84, 1995.
- [18] *Lentz, W.*: Numerische Suchverfahren als Hilfsmittel zur Ableitung von Handlungsstrategien aus bioökonomischen Simulationsmodellen. *Agrarwissenschaft*, 38, pp. 40-50, 1989.
- [19] *Mayrogianopoulos, G. N.*: Dynamic optimization of night temperature in relation to climate control in greenhouse. *Acta Horticulturae*, 287, pp. 109-116, 1991.
- [20] *Seginer, I.*: Expected usefulness of CO₂ enrichment and shading in a desert climate. *Acta Horticulturae*, 268, pp. 149-156, 1990.
- [21] *Tantau, H. J.*: Greenhouse climate control. Proceedings of the international seminar and British-Israel work- shop on greenhouse technology, Bet Dagan, Israel, 26 March-2 April 1990, pp. 91-101, 1990.

6.2.2 Simulation und Optimierung

- [22] *Chipperfield, A. J., Fleming, P. J. and Pohlheim, H.*: A Genetic Algorithm Toolbox for MATLAB. Proc. Int. Conf. Sys. Engineering, Coventry, UK, 6-8 Sept., 200-207, 1994.
- [23] *Dormand, J. R. and Prince, P. J.*: Practical Runge-Kutta Processes. *SIAM J. Sci. Stat. Comput.* 10, 977-989, 1989.
- [24] *Fehlberg, E.*: Klassische Runge-Kutta-Formeln fünfter und siebenter Ordnung mit Schrittweiten-Kontrolle. *Computing* 4, 93-106, 1969, Berichtigung in *Computing* 5, 184, 1970.
- [25] *Fehlberg, E.*: Klassische Runge-Kutta-Formeln vierter und niedrigerer Ordnung mit Schrittweiten-Kontrolle und ihre Anwendung auf Wärmeleitungsprobleme. *Computing* 6, 61-71, 1970.
- [26] *Fehlberg, E.*: Some Old and New Runge-Kutta Formulas with Step-size Control and Their Error Coefficients. *Computing* 34, 265-270, 1985.
- [27] *Fogel, D. B.*: An Introduction to Simulated Evolutionary Optimization. *IEEE Trans. on neural networks*, vol. 5, No. 1, 3-14, 1994.
- [28] *Fogel, L. J., Owens, A. J. and Walsh, M. J.*: Artificial Intelligence through Simulated Evolution. New York: John Wiley, 1966.
- [29] *Goldberg, D. E.*: Genetic Algorithms in Search, Optimization, and Machine Learning. Reading, Mass.: Addison-Wesley, 1989.
- [30] *Hairer, E., Norsett, S. P. and Wanner, G.*: Solving ordinary differential equations I. Berlin/Heidelberg: Springer, 1987.
- [31] *Hairer, E. and Wanner, G.*: Solving ordinary differential equations II. Berlin/Heidelberg: Springer 1991.

-
- [32] *Heinzel, G.*: Beliebige genau (Moderne Runge-Kutta-Verfahren zur Lösung von Differentialgleichungen). c't 8, 172-186, 1992.
- [33] *Holland, J. H.*: Adaptation in natural and artificial systems. Ann Arbor: The University of Michigan Press, 1975.
- [34] *Michalewicz, Z.*: Genetic Algorithms + Data Structures = Evolution Programs. Berlin, Heidelberg, New York: Springer, 1992.
- [35] *Mühlenbein, H. and Schlierkamp-Voosen, D.*: Predictive Models for the Breeder Genetic Algorithm: I. Continuous Parameter Optimization. Evolutionary Computation, 1 (1), 25-49, 1993.
- [36] *Philippow, E. u.a.*: Taschenbuch der Elektrotechnik, Band 2: Grundlagen der Informationstechnik. Berlin: Verlag Technik, 1977.
- [37] *Pohlheim, H.*: Ein genetischer Algorithmus mit Mehrfachpopulationen zur Numerischen Optimierung. at-Automatisierungstechnik 3 (1995), pp. 127-135, Berlin, 1995.
- [38] *Press, W. H.*: Numerical Recipes in C - The Art of Scientific Computing - Second Edition. Cambridge University Press, 1992
- [39] *Rechenberg, I.*: Evolutionsstrategie 94. Stuttgart: Frommann-Holzboog, 1994.
- [40] *Voigt, H.-M., Born, J. and Santibanez-Koref, I.*: Modelling and Simulation of Distributed Evolutionary Search Processes for Function Optimization. Proceedings of Parallel Problems Solving from Nature, pp. 373-380, volume 496 of Lecture Notes in Computer Science, Berlin, Heidelberg, New York: Springer, 1991.

6.2 Tabellen des Gewächshausmodelles

Tab. 4: Physikalische Konstanten

Größen	Symbol	Einheit	Wert
Gaskonstante des Wasserdampfes	RWS	J/(g·K)	0.46152
Spezifische Wärmekapazität der Luft	CPL	J/(g·K)	1.006
Spezifische Wärmekapazität des Wasserdampfes	CPD	J/(g·K)	1.8631
Spezifische Wärmekapazität des Wassers	CPW	J/(g·K)	4.1868
Verdampfungswärme von Wasser bei 20 ° C	VDW	J/g	2453
Verdampfungswärme von Wasser bei 0 ° C	VDW0	J/g	2501
Dichte der trockenen Luft bei 20 ° C und 760 Torr	DL	g/m ³	1204
Dichte von CO ₂ bei 20 ° C und 760 Torr	DC	g/m ³	1840
Sättigungsdampfdruck PSX in Pa für Wasser über Wasser bei einer Absoluttemperatur TAX in K: PSX = exp (a ₁ / TAX + a ₂ + a ₃ · TAX + a ₄ · TAX ² + a ₅ · log TAX)	PSX a ₁ a ₂ a ₃ a ₄ a ₅	Pa	-6094.4642 21.1249952 -0.02724555 0.0000168534 2.4575506

Tab. 5: Einflußgrößen

Größen	Symbol	Einheit
Dampfdichte	DDI	g/m ³
Ableitung von DDI nach der Zeit	DDDI	g/(m ³ ·s)
Dampfdruck	PDI	Pa
Dampfdrucksättigungsdefizit der Luft	SDI	hPa
Lufttemperatur	TEMI	°C
Ableitung von TEMI nach der Zeit	DTEMI	K/s
CO ₂ -Konzentration	CI	ppm
Ableitung von CI nach der Zeit	DCI	ppm/s

Tab. 6: Parameter des Gewächshauses

Größen	Symbol	Einheit	Wert
thermische Wirksamkeit der Globalstrahlung	KI		0.6

Transmissionsgrad des Gewächshauses	BQ		0.71
Mittlere Gewächshaushöhe	GH	m	3
Verhältnis aus Oberfläche und Grundfläche	GF		1.64
Wärmedurchgangskoeffizient bei Windstille	KW0	W/(m ² ·K)	3
Ableitung von $KW = KW0 + KW1 \cdot U$ nach der Windgeschwindigkeit	KW1	(W/(m ² ·K)) / (m/s)	0.2
Wärmeübertragungskoeffizient für den Wärmeübergang am Boden	KB	W/(m ² ·K)	3
Maximale Kondensatmenge an der Gewächshaushülle	KSM	g/m ²	25
Koeffizient zur Berechnung der Luftbefeuchtung	KR	1/Pa	0.0005

Tab. 7: Hilfsgrößen des Gewächshauses

Größen	Symbol	Einheit	Wert
Bestrahlungsstärke überm Bestand	I	W/m ²	BQ · IGLOB
PPFD über dem Bestand	IPHOT	μmol / (m ² ·s)	2.24 · I
Temperatur der Bodenoberfläche	TEMB	°C	20
Temperatur der Gewächshaushülle Winter (Oktober - April)	TEMG	°C	1/3 · TEMI + 2/3 · TEMA
Temperatur der Gewächshaushülle Sommer (Mai - September) nach	TEMG	°C	-2.71 + 0.00811 · IGLOB + 0.795 · TEMI + 0.289 · TEMA
Absoluttemperatur von TEMI	TAI	°K	TEMI + 273.15 K
Absoluttemperatur von TEMG	TAG	°K	TEMG + 273.15 K
Sättigungsdampfdruck für TEMI	PSI	Pa	siehe PSX (Tab. 4)
Sättigungsdampfdruck für TEMG	PSG	Pa	siehe PSX (Tab. 4)
Dampfdruck	PDX	Pa	DDX · TAX · RWS
Sättigungsdefizit	SDX	hPa	(PSX - PDX) / 100
Relative Luftfeuchte innen	FI	% r.F.	(PDI / PSI) · 100

6.3 Tabellen der Bilanzgleichungen

Tab. 8: Größen der Energiebilanz

Größen	Symbol	Einheit	Funktionen
Heizwärmestromdichte	Q	W/m ²	Stellgröße
Erwärmung durch	qglob	W/m ²	KI · I

Globalstrahlung			
Energiegehalt der Innenluft	HI	J/m ³	CPL · DL · TEMI + DDI · (VDW0 + CPD · TEMI)
Energiegehalt der Außenluft	HA	J/m ³	CPL · DL · TEMA + DDA · (VDW0 + CPD · TEMA)
Energieaustausch durch Luftwechsel	qluwe	W/m ²	LR · (1 / 3600) · (HI - HA)
Wärmedurchgang durch die Gewächshaushülle	qduga	W/m ²	GF · (KW0 + KW1 · U) · (TEMI - TEMA)
Wärmeübergang an der Bodenoberfläche	qboden	W/m ²	KB · (TEMI - TEMB)
Energieübertragung durch Kondensation bzw. Verdunstung an der Gewächshaushülle	qkonden	W/m ²	(1 / 3600) · VDW · kondverd
Wärmekapazität des Gewächshauses (Luft, Pflanzenbestand)	CG	J/(K·m ²)	GH · CPL · DL + A · KP · CPW

Tab. 9: Größen der Wasserdampfbilanz

Größen	Symbol	Einheit	Funktionen
Wasserzufuhr zur Luftbefeuchtung	RM	g/(m ² ·h)	Stellgröße
Transpiration des Bestandes	trans	g/(m ² ·h)	BLFL · n · trblfl (SDI, I, BLFL)
Luftbefeuchtung	lube	g/(m ² ·h)	KR · RM · (PSI - PDI)
Wasserdampfaustausch durch Luftwechsel	wadawe	g/(m ² ·h)	LR · (DDI - DDA)
Transportkoeffizient des Wasserdampfes	trpoko	m/h	$1.33 \cdot 3600 \cdot TEMI - TEMG ^{0.33} / (DL \cdot CPL)$
Kondensation bzw. Verdunstung an der Gewächshaushülle	konden	g/(m ² ·h)	trpoko · GF · (1/ (RWS · 0.5 · (TAI + TAG))) · (PDI - PSG)

Tab. 10: Größen der Kohlendioxidbilanz

Größen	Symbol	Einheit	Funktionen
CO ₂ -Massestromdichte (Anreicherung)	W	g/(m ² ·h)	Stellgröße

CO ₂ -Gaswechsel des Bestandes	gawe	g/(m ² ·h)	BLFL · n · 10 ⁻³ · gaweblfl (TEMI, SDI, CI, IPHOT, BLFL)
Austausch von CO ₂ durch Luftwechsel	kodiwe	g/(m ² ·h)	LR · DC · 10 ⁻⁶ · (CI - CA)

6.4 Tabellen des Pflanzenwachstumsmodelles

Tab. 11: Parameter des Pflanzenbestandes

Größen	Symbol	Einheit	Wert
Faktor zur Umrechnung der Blattfläche in Wasseräquivalent	KP	g/m ²	1000
Pflanzdichte	n	1/m ²	
Blattfläche je Pflanze	BLFL	dm ²	
Blattflächenindex	A		BLFL · n · 10 ⁻² Februar: 0.5 April: 1.5 Juni: 2.5
CO ₂ -Gaswechsel pro Blattfläche unter Standardbedingungen (TEMI = 25 °C, I = 300 W/m ² Globalstrahlung, CI = 600 ppm, SDI = 10 hPa)	P0	g/(dm ² ·h)	0.015 (0.015 - 0.03)
Transpiration pro Blattfläche unter Standardbedingungen (SDI = 10 hPa, I = 300 W / m ² Globalstrahlung, PDI / PSI = 0.6, CI = 600 ppm)	V0	g/(dm ² ·h)	1.0 (1.5 - 1.0)

6.5 Meteorologische Daten

Fig. 19: Tagesgang von Temperatur außen für Februar, April und Juni

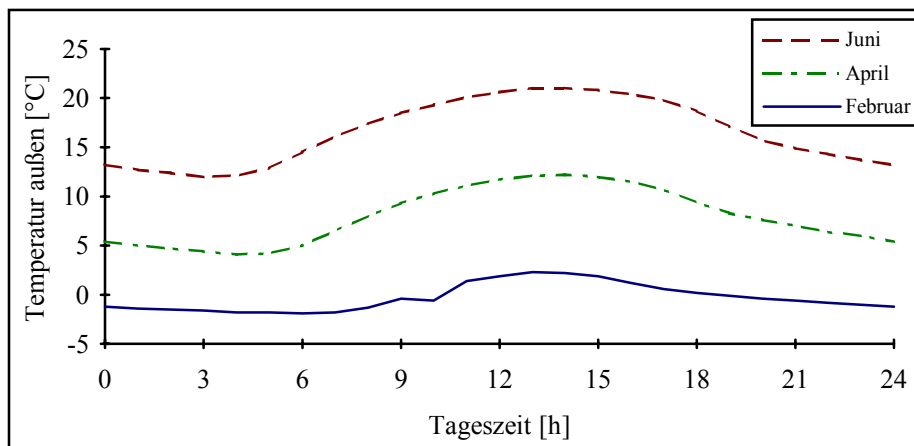


Fig. 20: Tagesgang von Feuchte außen für Februar, April und Juni

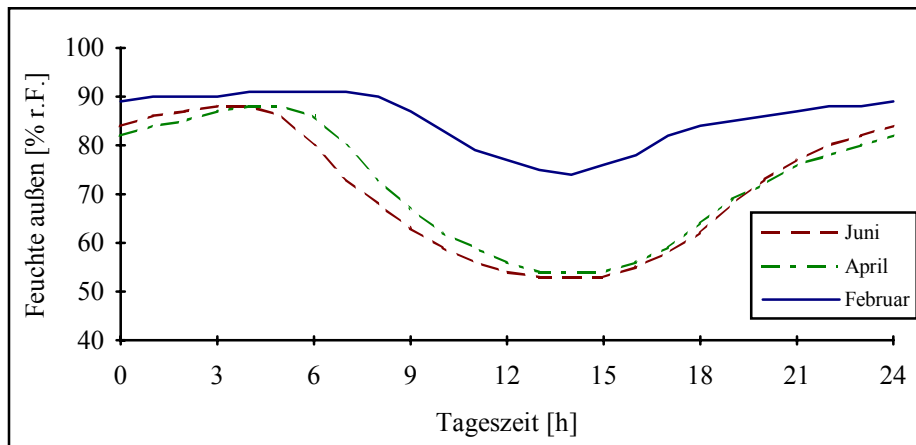


Fig. 21: Tagesgang von Globalstrahlung für Februar, April und Juni

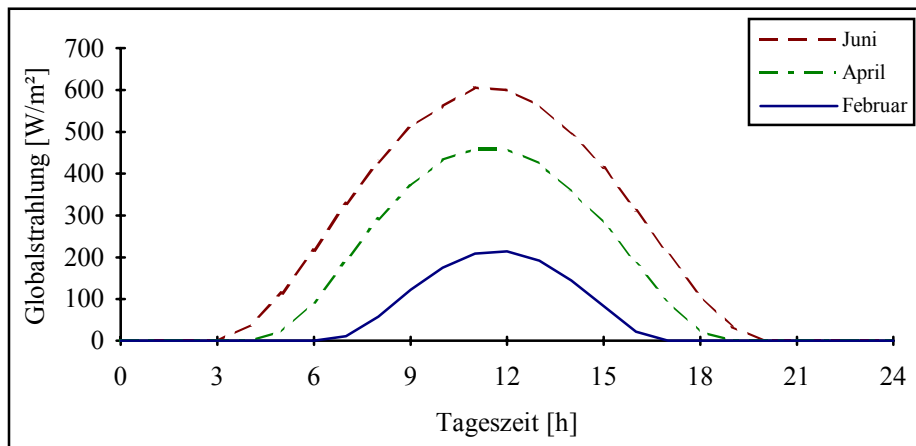
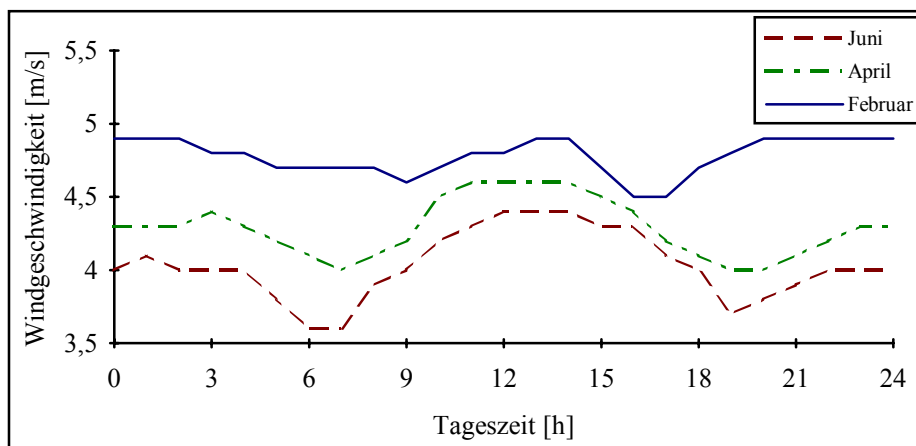


Fig. 22: Tagesgang von Windgeschwindigkeit außen für Februar, April und Juni



Für den Kohlendioxidgehalt außen wird ein konstanter Wert von $CA = 340$ ppm verwendet.

6.6 Grundlagen der Simulation

Das Modell des Pflanzenwachstums ist als System von Differentialgleichungen (DGL) gegeben. Die Simulation entspricht der Lösung dieses Differentialgleichungssystems (DGLS) über den interessierenden Zeitraum. Oft wird dies als Lösung eines Anfangswertproblems bezeichnet.

Mit den bekannten Werten der Variablen des DGLS zum Beginn des Zeitraums hat das DGLS eine exakte Lösung zu jedem späteren Zeitraum. Allerdings läßt sich diese exakte Lösung nur in den seltensten Fällen direkt ermitteln. Numerische Algorithmen (Integrationsverfahren) ermöglichen eine schnelle und genaue Lösung solcher Probleme .

6.6.1 Das EULERSche Polygonzugverfahren

Die einfachste und auch anschaulichste Methode ist das **EULERSche Polygonzugverfahren**.

Gegeben sei eine DGL 1. Ordnung $\dot{y} = f(t, y)$ mit dem Startpunkt $y(t_0)=y_0$. Die Funktion $\dot{y} = f(t, y)$ definiert in jedem Punkt die Steigung der Lösungskurve, die durch diesen Punkt hindurchgeht und damit auch für den Startpunkt. Für ein kurzes Zeitintervall (die Schrittweite h) wird die Lösungskurve durch ein Geradenstück angenähert. Dabei ist die Geradensteigung gleich der Steigung der Lösungskurve durch den Punkt. Damit ergibt sich folgende Formel: $y_{\text{neu}}=y_{\text{aktuell}}+h \cdot f(t_{\text{aktuell}}, y_{\text{aktuell}})$, die einen neuen Wert der DGL zur Zeit $t_{\text{neu}}=t_{\text{aktuell}}+h$ liefert, der als neuer Startpunkt verwendet werden kann und so weiter bis zum Ende des Zeitraumes.

Dieses Verfahren ist einfach und funktioniert gut für einfache DGL bzw. geringe Forderungen an die Genauigkeit der Lösung. Aber für höhere Forderungen an die Genauigkeit muß die Schrittweite sehr klein gewählt werden, um den Fehler durch die stückweise Linearisierung nicht zu groß werden zu lassen. Damit steigt der Aufwand durch die Auswertung der DGL, was vor allem bei aufwendigen DGLS zu einem starken Anstieg der benötigten Rechenzeit führt.

6.6.2 Das klassische RUNGE-KUTTA-Verfahren

Ebenfalls seit vielen Jahren ist das **klassische RUNGE-KUTTA-Verfahren 4. Ordnung** bekannt. Hierbei wird der neue Punkt der Lösungskurve nicht durch eine einfache Geradenapproximation ermittelt, sondern über die Berechnung von 4 Hilfsgrößen nach folgenden Gleichungen:

Equ. 10:

$$\begin{aligned} k_0 &= h \cdot f(t_{\text{aktuell}}, y_{\text{aktuell}}) \\ k_1 &= h \cdot f\left(t_{\text{aktuell}} + \frac{h}{2}, y_{\text{aktuell}} + \frac{k_0}{2}\right) \\ k_2 &= h \cdot f\left(t_{\text{aktuell}} + \frac{h}{2}, y_{\text{aktuell}} + \frac{k_1}{2}\right) \\ k_3 &= h \cdot f(t_{\text{aktuell}} + h, y_{\text{aktuell}} + k_2) \end{aligned}$$

Die erste Hilfsgröße (k_0) wird wie beim EULERSchen Polygonzugverfahren berechnet. Dann folgen zwei Auswertungen in der Mitte des betrachteten Intervalls, wobei jeweils das Ergebnis der vorigen Hilfsgrößenberechnung berücksichtigt wird. Die vierte Hilfsgröße

(k_3) wird am Ende des Intervalls ermittelt. Aus diesen 4 Werten errechnet sich der neue Punkt:

Equ. 11:

$$y_{\text{neu}} = y_{\text{aktuell}} + \left(\frac{1}{6}k_0 + \frac{1}{3}k_1 + \frac{1}{3}k_2 + \frac{1}{6}k_3\right)$$

Auch bei diesem Verfahren läßt sich durch die Wahl der Schrittweite das Ergebnis der geforderten Genauigkeit anpassen. Für eine hinreichend kleine Schrittweite kann die wahre Lösung beliebig genau angenähert werden.

Für eine Arbeit mit diesem und allen anderen Verfahren ist eine Abschätzung des aufgetretenen Fehlers pro Schritt und über den gesamten Zeitbereich notwendig. Der Fehler ist die Differenz aus berechnetem und exaktem Wert und abhängig von der Schrittweite (h) und der Ordnung (q) des Verfahrens:

Equ. 12:

$$y_{\text{berechnet}} - y_{\text{exakt}} = O(h^{q+1}) \Leftrightarrow \Delta y_{\text{Schritt}} \approx h^{q+1} \text{ und } \Delta y_{\text{gesamt}} \approx h^q \text{ für } h \rightarrow 0$$

Das EULERSche Polygonzugverfahren ist 1. Ordnung. Durch Halbierung der Schrittweite wird eine Fehlerverkleinerung pro Schritt auf $(\frac{1}{2}h)^{q+1} \approx \frac{1}{2}\Delta y_{\text{Schritt}}$ erreicht. Mit anderen Worten, durch Verdopplung der Schrittzahl wird ein doppelt genaues Ergebnis erreicht.

Für das klassische RUNGE-KUTTA-Verfahren 4. Ordnung sieht diese Bilanz etwas besser aus: $(\frac{1}{2}h)^{q+4} \approx \frac{1}{16}\Delta y_{\text{Schritt}}$. Durch eine Verdopplung der Schrittzahl wird ein (im Idealfall) 16mal genaueres Ergebnis erreicht. Dies zeigt, warum das klassische RUNGE-KUTTA-Verfahren so populär ist und auch heute an vielen Stellen ein gut geeignetes Verfahren ist.

Ein Problem bleibt noch. Die absolute Größe des Fehlers pro Schritt kann immer noch nicht abgeschätzt werden (die Abschätzung des Gesamtfehlers ist Gegenstand der aktuellen Forschung, siehe [23]). Bei einem oft verwendeten Verfahren wird der letzte Schritt noch einmal mit der halben Schrittweite durchgeführt, die Differenz der beiden ermittelten Werte gebildet und mit einer vorgegebenen Fehlerschranke verglichen. Ist die Differenz kleiner, war der erste Schritt gültig und es kann mit derselben oder sogar einer größeren Schrittweite weitergerechnet werden. War die Differenz größer als die Fehlerschranke, so muß der Schritt mit einer kleineren Schrittweite wiederholt werden.

Mit diesem einfachen Algorithmus ist eine leichte Steuerung der Schrittweite in Abhängigkeit des Fehlers pro Schritt im Vergleich zu einer vorgegebenen Fehlerschranke möglich. Einen Nachteil hat die Einfachheit aber: sie hat eine Verdopplung des Rechenaufwandes zur Folge. Dies kann zwar am Ende immer noch effektiver sein, als ohne Schrittweitensteuerung zu arbeiten, aber im folgenden werden Verfahren gezeigt, die eine effektivere Berechnung des entstandenen Fehlers mit einer hohen Ordnung verbinden.

6.6.3 Eingebettete RUNGE-KUTTA-Verfahren

Bei den eingebetteten Verfahren, die seit den 60'er Jahren entwickelt wurden, werden zwei RUNGE-KUTTA-Formeln unterschiedlicher Ordnung miteinander kombiniert. Der Wert der Formel mit der höheren Ordnung und damit der genauere von beiden wird als neuer Punkt weiterverwendet. Die Differenz mit dem Wert der Formel niedrigerer Ordnung wird als Maß für den Fehler für die Schrittweitensteuerung benutzt.

Die Koeffizienten werden so gewählt, daß die Berechnungen für die Formel niedrigerer Ordnung bei der höheren Ordnung wiederverwendet werden können. Dies bedeutet, daß z.B. für ein eingebettetes Verfahren 6.(5.) Ordnung 7 Hilfsgrößen für die Formel 6. Ordnung und

mindestens 1 zusätzliche Hilfsgröße für die Kontrollformel 5. Ordnung, insgesamt 8 Hilfsgrößen ($s=8$, Anzahl der Hilfsgrößen) benötigt werden [24].

Die Berechnungen der RUNGE-KUTTA-Verfahren lassen sich in allgemeiner Form schreiben. Für die Hilfsgrößen ergibt sich:

Equ. 13:

$$k_i = f(t_{\text{aktuell}} + c_i h_{\text{aktuell}}, y_{\text{aktuell}} + h_{\text{aktuell}} \sum_{j=0}^{i-1} a_{ij} k_j) \quad \text{mit } c_i = \sum_{j=0}^{i-1} a_{ij} \quad \text{und } i = 0, \dots, s-1$$

Der neue Punkt berechnet sich nach:

Equ. 14:

$$y_{\text{neu}} = y_{\text{aktuell}} + h_{\text{aktuell}} \cdot \sum_{i=0}^{s-1} b_i k_i \quad (\text{Ordnung } q)$$

Die Gleichung zur Fehlerberechnung ist bis auf die Koeffizienten identisch mit der zur Berechnung des neuen Punktes:

Equ. 15:

$$\hat{y}_{\text{neu}} = y_{\text{aktuell}} + h_{\text{aktuell}} \cdot \sum_{i=0}^{s-1} \hat{b}_i k_i \quad (\text{Ordnung } p)$$

Die einzelnen Verfahren unterscheiden sich nur in der Ordnung und den Koeffizienten. Auch das klassische RUNGE-KUTTA-Verfahren läßt sich in dieser Form angeben.

Tab. 12:

a_{ij}			b_i	c_i	
0	0	0	$\frac{1}{6}$	0	mit $q = 4$ und $s = 4$; kein $p \Rightarrow$ kein \hat{b}
$\frac{1}{2}$	0	0	$\frac{1}{3}$	$\frac{1}{2}$	
0	$\frac{1}{2}$	0	$\frac{1}{3}$	$\frac{1}{2}$	
0	0	1	$\frac{1}{6}$	1	

Eines der bekanntesten Formelpaare wurde von FEHLBERG 1969 [24] vorgestellt, von ihm als RK5(6) bezeichnet. (Bei diesem wird der Wert der Formel mit der niedrigeren Ordnung als neuer Punkt weiterverwendet.)

Tab. 13:

a_{ij}						b_i	\hat{b}_i	c_i	
0						$\frac{31}{384}$	$\frac{7}{1408}$	0	mit $q = 5$ $p = 6$ $s = 8$
$\frac{1}{6}$						0	0	$\frac{1}{6}$	
$\frac{4}{75}$	$\frac{16}{75}$					$\frac{1125}{2816}$	$\frac{1125}{2816}$	$\frac{4}{15}$	
$\frac{5}{6}$	$-\frac{8}{3}$	$\frac{5}{2}$				$\frac{9}{32}$	$\frac{9}{32}$	$\frac{2}{3}$	
$-\frac{8}{5}$	$\frac{144}{25}$	-4	$\frac{16}{25}$			$\frac{125}{768}$	$\frac{125}{768}$	$\frac{4}{5}$	
$\frac{361}{320}$	$-\frac{18}{5}$	$\frac{407}{128}$	$-\frac{11}{80}$	$\frac{55}{128}$		$\frac{5}{66}$	0	1	
$-\frac{11}{640}$	0	$\frac{11}{256}$	$-\frac{11}{160}$	$\frac{11}{256}$	0		$\frac{5}{66}$	0	
$\frac{93}{640}$	$-\frac{18}{5}$	$\frac{803}{256}$	$-\frac{11}{160}$	$\frac{99}{256}$	0	1	$\frac{5}{66}$	1	

In den letzten Jahren wurden neue RUNGE-KUTTA-Formelpaare vorgestellt, die auf Effizienz und kleine Fehler-Koeffizienten getrimmt wurden, um so mit einer möglichst großen Schrittweite arbeiten zu können. Mehrere Arbeiten veröffentlichten DORMAND und PRINCE zu diesem Thema. Als Beispiel sei ihr Formelpaar RK5(4)7 aufgeführt:

Tab. 14:

a_{ij}						b_i	\hat{b}_i	c_i	
0						$\frac{35}{384}$	$\frac{5179}{57600}$	0	
$\frac{1}{5}$						0	0	$\frac{1}{5}$	
$\frac{3}{40}$	$\frac{9}{40}$					$\frac{500}{1113}$	$\frac{7571}{16695}$	$\frac{3}{10}$	mit $q = 5$
$\frac{44}{45}$	$-\frac{56}{15}$	$\frac{32}{9}$				$\frac{125}{192}$	$\frac{393}{640}$	$\frac{4}{5}$	$p = 4$
$\frac{19372}{6561}$	$-\frac{25360}{2187}$	$\frac{64448}{6561}$	$-\frac{212}{729}$			$-\frac{2187}{6784}$	$-\frac{92097}{339200}$	$\frac{8}{9}$	$s = 7$
$\frac{9017}{3168}$	$-\frac{355}{33}$	$\frac{46732}{5247}$	$\frac{49}{176}$	$-\frac{5103}{18656}$		$\frac{11}{84}$	$\frac{187}{2100}$	1	
$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	0	$\frac{1}{40}$	1	

Weitere Formelpaare finden sich in [24], [25], [26], eine gute Übersicht geben [32] und vor allem [30].

6.7 Grundlagen der Dynamischen Optimierung

Im folgenden sollen die Grundlagen für die Optimierung dargestellt werden ([36]). Im ersten Abschnitt erfolgt eine mathematische Beschreibung dynamischer Systeme zusammen mit Beschränkungen und Gütekriterien. Danach folgt eine kurze Erläuterung der Funktionsweise von deterministischen Suchverfahren ohne/mit Verwendung von Gradienteninformationen.

Für das oben dargestellte Modell soll der Verlauf einer Steuerung unter Beachtung der Zustandsgleichungen ermittelt werden. Damit kommt die dynamische Optimierung zur Anwendung, d.h., es wird eine optimale Steuerung in einem bestimmten Zeitintervall gesucht.

6.7.1 Systembeschreibung

Bei dem vorliegenden Modell handelt es sich um ein kontinuierliches System mit konzentrierten Parametern.

Das System wird durch ein Differentialgleichungssystem wie folgt beschrieben:

$$\begin{bmatrix} \dot{q}_1(t) \\ \vdots \\ \dot{q}_n(t) \end{bmatrix} = f \left(\begin{bmatrix} q_1(t) \\ \vdots \\ q_n(t) \end{bmatrix}, \begin{bmatrix} u_1(t) \\ \vdots \\ u_m(t) \end{bmatrix} \right) \quad \begin{array}{l} q(t_0) = q_0: \text{ Startvektor} \\ q(t): \text{ Zustandsvektor} \\ u(t): \text{ Steuervektor} \end{array}$$

Für die Komponenten gelten diese Bedingungen:

- Die Steuerung $u(t)$ muß im gesamten betrachteten Zeitintervall $[t_0, t_1]$ eine integrierbare Funktion sein, damit ist nur eine endliche Anzahl von Sprüngen zugelassen und zu jedem Zeitpunkt $t \in [t_0, t_1]$ ist $u(t) \in \mathbb{R}^m$.
- Der Zustandsvektor $q(t)$ muß im gesamten betrachteten Zeitintervall $[t_0, t_1]$ eine stetige Funktion sein und zu jedem Zeitpunkt $t \in [t_0, t_1]$ ist $q(t) \in \mathbb{R}^n$.

6.7.2 Beschränkungen

Während der Optimierung können verschiedene Arten von Beschränkungen auftreten: Endzustands-, Steuerungs- und Zustandsbeschränkungen. Diese lassen sich in allgemeiner Form angeben:

- Endzustand: $a_i[q(t_1)] = 0 \quad i = 1, \dots, k_e \quad k_e \leq n$
- Steuerung: $b_i[u(t)] \leq 0 \quad i = 1, \dots, k_u$
zulässiger Steuerbereich: $U_B = \{u(t): b_i[u(t)] \leq 0, \quad i = 1, \dots, k_u\} \in \mathbb{R}^m$
- Zustand: $c_i[q(t)] \leq 0 \quad i = 1, \dots, k_q$
zulässiger Zustandsbereich: $Q_B = \{q(t): c_i[q(t)] \leq 0, \quad i = 1, \dots, k_q\} \in \mathbb{R}^n$

Dabei kann man für jede Beschränkungsart Spezialfälle angeben.

Endzustandsbeschränkungen:

- freies Ende (keine Beschränkungen): $q(t_1) \in \mathbb{R}^n$
- festes Ende (Punkt als Zielmenge): $q(t_1) = q_1$
- gebundenes Ende

Steuerungsbeschränkungen:

- Komponentenbeschränkungen: $u_{i_{\min}} \leq u_i(t) \leq u_{i_{\max}}; \quad i = 1, \dots, k_u / 2$
- Hyperebenenbeschränkungen: $\alpha_i u(t) + \beta_i \leq 0; \quad i = 1, \dots, k_u$
- Normbeschränkungen: $\|u(t)\| \leq b$

Für die Zustandsbeschränkungen kann man dieselben Spezialfälle wie für die Steuerungsbeschränkungen angeben.

Alle Beschränkungen für ein System müssen miteinander verträglich sein. Die von den Steuerungen $u(t) \in U_B$ erzeugte Menge der Zustände in \mathbb{R}^n muß für jedes $t \in [t_0, t_1]$ mit dem zulässigen Zustandsbereich Q_B einen nichtleeren Durchschnitt bilden. Die Steuerungen, die unter Beachtung der Zustandsgleichungen alle gegebenen Beschränkungen erfüllen, sind die zulässigen Steuerungen, unter denen die optimale Steuerung $u^*(t)$ auszuwählen ist.

6.7.3 Gütekriterien

Als Gütekriterium dient ein entsprechend der Aufgabe zu definierendes Zielfunktional:

- BOLZA-Form:
$$I = f^1[q(t_1)] + \int_{t_0}^{t_1} f^0[q(t), u(t)] dt$$

Das Ziel der Optimierung ist die Minimierung des Zielfunktional I . Dabei bewertet $f^1[q(t_1)]$ den zu erreichenden Endzustand, wobei diese Bewertung nur dann einen Sinn hat, wenn $q(t_1)$ nicht fest ist. Der zweite Term bewertet Zustand und Steuerung über den gesamten Optimierungszeitraum.

Neben der obigen Form des Zielfunktional gibt es zwei weitere, die jeweils einen der Teile der BOLZA-Form enthalten:

- LAGRANGE-Form:
$$I = \int_{t_0}^{t_1} f^0[q(t), u(t)] dt$$
- MAYER-Form:
$$I = f^1[q(t_1)]$$

Diejenige zulässige Steuerung, die das gewählte Zielfunktional minimiert, ist die optimale Steuerung $u^*(t)$.

6.8 Optimierungsverfahren

6.8.1 Deterministische Suchverfahren

Es ist relativ einfach, eine Funktion, die nur von einer Variablen abhängt zu minimieren. Hier seien die Suche nach dem Goldenen Schnitt bzw. durch Parabolische Interpolation genannt. Wenn man von einem Punkt P im N -dimensionalen Raum startet, so kann jede Funktion von N Variablen $f(P)$ entlang einiger Richtungen n mit den eindimensionalen Verfahren minimiert werden. Verschiedene Verfahren differieren nur darin, wie sie die neue Richtung aussuchen, in die der nächste Versuch stattfindet. Dabei gibt es zwei Ziele: das Verfahren enthält einige sehr gute Richtungen, die durch ein langes, schmales Tal führen oder das Verfahren enthält eine Anzahl von Richtungen mit der speziellen Eigenschaft, daß die Minimierung entlang einer dieser Richtungen nicht von nachfolgenden Minimierungen

in einer anderen Richtung verdorben wird, wodurch ein ewiges Durchlaufen der Richtungen verhindert werden kann.

Dieses Konzept der unabhängigen Richtungen wird als Methode der Konjugierten Richtungen bezeichnet. Man stelle sich vor, in einer Richtung w ein Minimum erreicht zu haben und bewegt sich jetzt in einer neuen Richtung v weiter. Die Bedingung, daß die Bewegung in Richtung v nicht die Minimierung in Richtung w verdirbt, ist nur dann erfüllt, wenn der Gradient rechtwinklig auf w steht, mit anderen Worten, die Änderung des Gradienten rechtwinklig zu w ist. Wenn dies für die Vektoren w und v gilt, so sind diese konjugiert. Gilt diese Bedingung paarweise für alle Mitglieder eines Satzes von Vektoren, so werden diese als ein konjugierter Satz bezeichnet.

Bei einer schrittweisen Richtungsminimierung einer Funktion entlang eines konjugierten Satzes von Richtungen braucht man keine dieser Richtungen noch einmal zu minimieren. Für ein Verfahren ist es also günstig, einen Satz von N linear unabhängigen, wechselseitig konjugierten Richtungen zu liefern. Denn damit würde eine einmalige Abarbeitung der N Richtungsminimierungen direkt zum Minimum führen, wenn die Funktion von quadratischer Form ist. Für Funktionen, die keine exakte quadratische Form haben, wäre dieser erreichte Punkt nicht das exakte Minimum. Aber durch wiederholtes Abarbeiten der N Richtungsminimierungen wird das Minimum mit quadratischer Konvergenz erreicht.

POWELL entdeckte als erster eine Methode, um einen Satz von N untereinander konjugierten Richtungen zu erhalten ([38]). Diese Methode erfordert $N(N+1)$ Richtungsminimierungen für eine exakt quadratische Funktion. Diese Anzahl der Richtungsminimierungen erscheint auf den ersten Blick sehr hoch, allerdings darf an dieser Stelle nicht übersehen werden, daß nur die Funktion selbst benötigt wird.

6.8.2 Deterministische Suchverfahren unter Verwendung des Gradienten

Wenn nun die Möglichkeit der Berechnung des Gradienten der Funktion besteht, welche Vorteile bringt dies?

Dazu muß etwas genauer auf die Anzahl der notwendigen Berechnungen für die beiden Varianten geschaut werden. Die Anzahl der unbekannt Parameter ist von Ordnung N^2 . Wird einer dieser Parameter geändert, so kann dies den Ort des Minimums verschieben. Deshalb kann man nicht erwarten, das Minimum zu finden, bevor man eine gleichwertige Informationsmenge gesammelt hat, von Ordnung N^2 . Bei der Methode von POWELL wird diese Information durch N^2 einzelne Richtungsminimierungen gesammelt, wovon jede mehrere Auswertungen der Funktion erfordert.

Jede Ermittlung des Gradienten bringt N neue Komponenten. Wenn man diese Informationen günstig einsetzt, braucht man nur noch N einzelne Richtungsminimierungen durchzuführen, um das Minimum der Funktion zu ermitteln.

Ein einfaches Verfahren zur Einbeziehung des Gradienten könnte wie folgt aussehen: Begonnen wird bei einem Startpunkt P_0 . So oft wie nötig wird von P_i aus in Richtung $-\nabla f(P_i)$, entgegengesetzt zum Gradienten, eine Richtungsminimierung durchgeführt und so der neue Punkt P_{i+1} erreicht. Oft wird diese Variante als Gradientenverfahren bezeichnet.

Wie schon weiter oben angeführt ist es günstig nicht in Richtung des neuen Gradienten zu gehen, sondern in eine Richtung, die konjugiert zum vorigen Gradienten und zu allen vorhergehenden Richtungen ist. Verfahren, die solche Richtungen liefern, werden als

Verfahren der Konjugierten Gradienten bezeichnet. Das Verfahren der Konjugierten Gradienten arbeitet mit zwei Folgen von Vektoren (Gradient g und Suchrichtung s):

$$g_{i+1} = g_i - \lambda_i A \cdot s_i \quad s_{i+1} = g_{i+1} + \beta_i s_i \quad i = 0, 1, 2, \dots$$

mit dem Startvektor $g_0 = -\nabla f(P_0)$ und $s_0 = g_0$. Diese Vektoren erfüllen die Orthogonalitätsbedingungen und sind konjugiert zueinander:

$$g_i \cdot g_j = 0 \quad s_i \cdot A \cdot s_j = 0 \quad g_i \cdot s_j = 0 \quad i \neq j$$

Die skalaren Werte λ_i und β_i (Streckfaktor) sind gegeben durch:

$$\lambda_i = \frac{g_i \cdot g_i}{s_i \cdot A \cdot s_i} = \frac{g_i \cdot s_i}{s_i \cdot A \cdot s_i} \quad \beta_i = \frac{g_{i+1} \cdot g_{i+1}}{g_i \cdot g_i}$$

Angenommen, man kennt die HESSE-Matrix A . Dann könnte man mit den Gleichungen $g_{i+1} = g_i - \lambda_i A \cdot s_i$ und $s_{i+1} = g_{i+1} + \beta_i s_i$ Schritt für Schritt konjugierte Richtungen s_i finden, in deren Richtung man dann die Richtungsminimierung durchführt. Nach N solchen Minimierungen hätte man das Minimum der quadratischen Form erreicht. Leider ist A unbekannt.

Es gibt aber ein Theorem, das dieses Problem elegant löst: Ausgangspunkt ist der Gradient für einen Punkt P_i mit $g_i = -\nabla f(P_i)$. Angenommen, man geht von P_i aus in Richtung s_i zum lokalen Minimum von f am Punkt P_{i+1} und setzt $g_{i+1} = -\nabla f(P_{i+1})$. Dann ist g_{i+1} derselbe Vektor, den man durch die Gleichung $g_{i+1} = g_i - \lambda_i A \cdot s_i$ berechnet hätte, aber ohne A zu kennen.

Damit hat man eine Basis für einen Algorithmus, der weder die Kenntnis der HESSE-Matrix noch den Speicherplatz zu ihrer Verwaltung benötigt. Eine Folge von konjugierten Richtungen s_i wird nur durch Richtungsminimierungen und die Auswertung des Gradienten aufgebaut.

Aus $s_{i+1} = g_{i+1} + \beta_i s_i$ ist leicht zu ersehen, daß für $\beta_i = 0$ die Verfahren der Konjugierten Gradienten in das Gradientenverfahren übergehen. Dies wird außerdem verwendet, um nach N Minimierungsschritten das Verfahren mit einem Gradientenschritt neu zu starten. Damit kann ein Außer-Tritt-Kommen des Algorithmus verhindert werden, bzw., wenn dies auftrat, die Minimierung neu gestartet werden.

Die Suchverfahren unter Verwendung des Gradienten sind sehr leistungsfähig - wenn sie angewendet werden können. Die Aufstellung der Hamiltonfunktion, Berechnung der Kozustände und des Gradienten für dynamische Systeme ist kompliziert und oftmals kaum möglich.

Daher werden für diese Systeme oft die deterministischen Suchverfahren ohne Verwendung des Gradienten eingesetzt (z.B. konjugierte Richtungen nach POWELL). Sie sind einfacher anzuwenden. Allerdings erfordern diese aber gerade für Zielfunktionen, die nicht quadratisch sind, einen sehr hohen Rechenaufwand. Zudem sind diese Verfahren lokale Optimierungsverfahren und scheitern bei multimodalen Systemen fast immer.

Eine mögliche Lösung stellt die Anwendung Evolutionärer Algorithmen dar.