

# 施設園芸における栽培管理のMachine Learningによる自動化に関する基礎的研究

|       |            |
|-------|------------|
| 誌名    | 農業気象       |
| ISSN  | 00218588   |
| 著者    | 蔵田, 憲次     |
| 巻/号   | 44巻3号      |
| 掲載ページ | p. 181-186 |
| 発行年月  | 1988年12月   |

# 施設園芸における栽培管理のMachine Learning による自動化に関する基礎的研究

蔵 田 憲 次

(東京大学農学部農業工学科)

A Fundamental Study on Automation  
of Greenhouse Crop Management by Machine Learning

Kenji KURATA

(Department of Agricultural Engineering, University of Tokyo,  
Yayoi 1-1-1, Bunkyo-ku, Tokyo 113, Japan)

## 1. はじめに

近年、わが国では温室のコンピュータ制御装置はある程度普及した。しかし、施設園芸全体と対比すると、その普及率はまだ低い水準にある(たとえば、日本施設園芸協会, 1986)。その理由には、設定温度などの入力作業が煩雑で使い難いこと、制御アルゴリズムが固定しているため、必ずしも栽培者の要求に答え得ていないことなどが考えられる。

栽培管理には、地下水位が高いとか、日陰になりやすいなどの局地的な条件によって異なる個別的な行為が多くある。したがって、このような個別的な栽培管理に共通する一般的な制御アルゴリズムが存在するか否かは、検討を要する。また個別的な管理をコンピュータに行なわせる場合、栽培者が容易に必要なデータなどをコンピュータに入力できるようなシステム構成が必要である。その場合の難題は、栽培者は過去の経験などから、局地的な条件に適した栽培方法を体得してはいるが、栽培者自身が必ずしもその管理方法を客観的に意識化しているとは限らないことである。

そこで、(温室内)気象条件等の栽培管理を規定する条件と管理行為の両者を自動計測し、その中から栽培管理の原則(以後、栽培管理ルール、あるいは単にルールとよぶ)を抽出し、そのルールによって温室を自動制御するシステムの開発が望まれる。そのようなシステムの開発によって、機械が栽培者の方法に近い栽培管理を自動的にこなすことが可能となり、栽培者に余計な作業負担をかけることなく省力化が可能となる。このシステムの

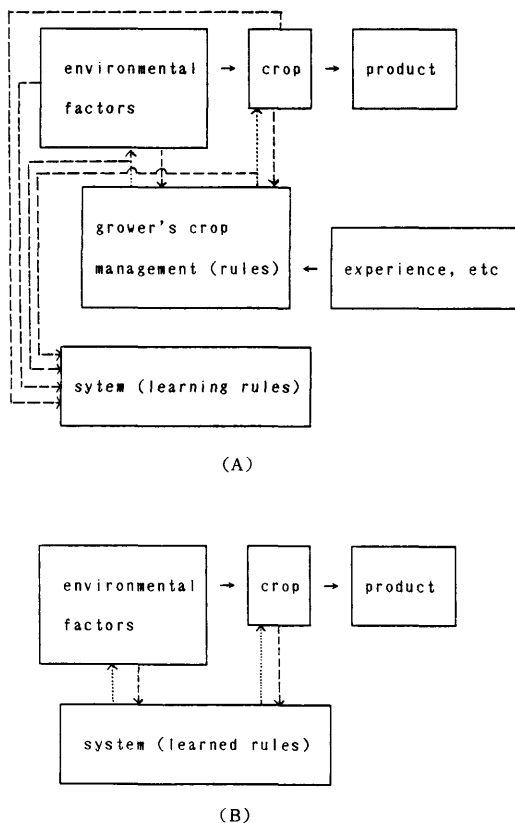


Fig. 1. A system we aim at, which automatically imitates grower's crop management. A: learning stage, B: managing stage, dotted line: act on, broken line: measured by, felt by.

概念図を Fig. 1 に示した。

システムの中核となるのは、計測値からルールを抽出する学習機能である。このような学習機能は、温室の自動制御に応用することにとどまらず、栽培管理ルールそのものの分析、とくに篤農家といわれる栽培者のルールの収集・分析にも一部応用可能であろう。本報告では、施設園芸における栽培管理ルールについて考察し、開発した学習アルゴリズムの可能性と問題点について報告する。なお、開発した学習アルゴリズムの詳細については、紙面の関係で別稿(Kurata, 1988)に譲る。

## 2. 栽培管理ルールの特徴と本報告の範囲

### 2.1 栽培管理ルール

まず、栽培管理ルールの特徴について考察する。栽培者が温室の天窓を開閉する例について考える。栽培者が、室温が少々高すぎると感じて、天窓を開ける場合、栽培管理ルールは次のように記述できる。

$$T > T_0 \text{ | } \text{Ivent} = 1 \quad (1)$$

ここで、 $T$  は温室内気温である。 $T_0$  はあらかじめ設定された（栽培者の意識内にある栽培管理ルールによって決っている）気温である。 $T$ 、 $T_0$  はより正確には、栽培者の感じた気温というべきである。 $T_0$  のようにある操作を行うか、行わないかの判断の基準となる値を基準値と呼ぶ。また、 $\text{Ivent}$  は、天窓を開けるととき 1、そうでないとき 0 となる関数である。| の左辺は、「もし～ならば」の条件部を表わし、右辺は「そのときだけ～する」の結論部を表わすものとする。すなわち、| は「もし～ならば、そのときだけ～する」の関係を表わすものとする。

次に、栽培者が、室温が少々高すぎるか、少し蒸し暑すぎると感じて、天窓を開ける場合を考える。蒸し暑さは、ここでは仮に、気温  $T$ 、相対湿度  $q$ 、日射量  $R$  の関数とする。すなわち、蒸し暑さを  $H$  とすると

$$H = f(T, q, R) \quad (2)$$

また、蒸し暑さの基準値を  $H_0$  とする（正確には、 $H$  と  $H_0$  は、栽培者の感じた蒸し暑さとその基準値）。上の栽培管理ルールは次のようになる。

$$(T > T_0) \text{ OR } (H > H_0) \text{ | } \text{Ivent} = 1 \quad (3)$$

ここで、OR は論理和を表わす。

次に、栽培者が作物を見て萎れそうだと感じて灌水する場合を考える。ただし、灌水は、午前中にしか行わないものとする。このルールを表わすには、栽培者がみた、萎れそうだという作物の状態を定義する関数が必要である。また、この場合の学習を可能にするには、作物の状態を自動計測する機能をもつ画像解析装置が必要であ

る。そのような機能はまだ未開発であり今後の開発が期待される(蔵田・大原, 1988 を参照)。ここでは作物の萎れそうだという状態は定義でき、計測可能なものと仮定する。作物の萎れそうだという状態を表わす関数を  $p$  で、その灌水の基準値を  $p_0$  で表わすと、上のルールは、次のように表わすことができる。

$$(p > p_0) \text{ AND } (6 < t < 12) \text{ | } \text{Iirri} = 1 \quad (4)$$

ここで、AND は論理積を、 $t$  は 1 日のうちの時刻を表わし、午前は 6 時から 12 時までとする。また、 $\text{Iirri}$  は灌水する場合 1 に、しない場合 0 となる関数である。

ここで、以上の諸例に基づき、次の各用語を定義する。

- 1) 環境関数：気象条件や土壌含水率などの環境条件を表わし、直接計測可能な関数(上例の  $T$ ,  $q$ ,  $R$ )。
- 2) 作物関数：作物の状態を表わし、直接計測可能な関数(上例の  $p$ )。
- 3) 時間関数：時間に関係する関数(上例の  $t$ )。
- 4) 派生関数：環境関数、作物関数あるいは、時間関数から導出される関数(上例の  $H$ )。
- 5) 条件関数：環境関数、作物関数、時間関数、派生関数の総称。
- 6) 操作関数：操作をするとき 1、しないとき 0 の値をとる関数(上例の  $\text{Ivent}$ ,  $\text{Iirri}$ )。
- 7) 基準値：条件関数の比較の対象となる値(上例の  $T_0$ ,  $H_0$ ,  $p_0$ )。
- 8) 項：条件関数とその基準値との大小関係の比較(上例の  $(T > T_0)$  など)。

式(1)から(4)の諸例は、栽培管理ルールのいくつかの特徴を示している。それらを基礎に、栽培管理ルールの特徴を列挙すると次のようになる。これらの中には、それが厳密に成り立つかどうかは、さらに詳細な検討を要するものもある。したがって、以下に述べる諸点は、栽培管理ルールの一般的特徴というよりは、本研究の基本的な仮定とみなすべきものである。

1) 栽培管理ルールは、「もし～ならば、そのときだけ～である」の形で記述できる。

2) 条件部は、条件関数の値とその基準値との大小関係の比較の項よりなる。すなわち、数値の比較である。この点は、カテゴリ間の関係に重点を置く Machine Learning と対照的である。たとえば、アーチの概念の学習では、柱やはりというカテゴリおよびそれらの間の位置関係が問題となる(たとえば、Winston, 1975)。

3) 条件部は、数値の大小関係を表わす項 1 つ、あるいは複数個からなる。複数個の場合、それらは AND あるいは OR の関係で結ばれている。

4) 結論部は、1 つの操作関数からなる。すなわち、

結論部では1か0以外の数値は扱わない。栽培管理はいくつかの操作を含む。操作の種類ごとに、栽培管理ルールを定義しているから、ルールの結論部が1つということとは、扱う操作も1つということとは異なる。

結論部は、操作関数の形で表わせるということについては議論が必要である。施設園芸における栽培管理では、ある操作を行うか、行わないか、という決定場面が多い。たとえば、灌水、換気窓の開閉、カーテンの開閉、薬剤散布、摘花、ホルモン処理、収穫などである。しかし、換気窓をどの程度開けるか、何mm灌水するか、というように、施設園芸での栽培管理には、単純に「行うか、行わないか」では処理できない要素があることは否定できない。しかし、多くの栽培管理操作は、「行うか、行わないか」の形をとる。また、0と1以外の数値をとる関数も操作関数に変換できる場合がある。たとえば、「何mm灌水するか」という問題は、灌水速度は一定であると仮定すると、「灌水を続けるか、止めるか」という問題に変換される。そこで、本研究では、第一歩として、栽培管理操作は、操作関数で記述できると仮定する。

5) 条件部には、あいまい (Fuzzy) な要素が関係する。上で述べた諸例の条件部にてきた環境関数や作物関数は、栽培者が感じたり、見て判断したものである。したがって、これらには、人間の感覚のもつあいまいさが混入している。同様に、派生関数や時間関数の定義にもあいまいさが混入している。時間そのものには曖昧さは関係しないが、その基準値はあいまいである。たとえば、夕方という概念をあいまいさ抜きで定義することは難しい。このように、栽培管理ルールには、あいまいさがさまざまな形で関係している。

## 2.2 要求される学習機能

要求される学習機能に関連して、次の3点について考察する。

1) 条件関数は、互いに独立とは限らない。たとえば、日射が強ければ、それだけ温室内に透入する日射量も多いから、温室内気温は上昇する。また、作物関数も、環境関数や時間関数と強い相関を持つことが予想される。たとえば、作物の萎れは、前回の灌水からの時間、日射量、湿度、土壌含水率などと相関があろう。このように、条件関数は相互に相関している場合が多い。

2) 条件関数は、操作関数の影響を受ける場合もある。すなわち、条件関数は人為的操作が不可能な関数とは限らない。栽培管理は、条件関数を変更する目的で行われることが多い。たとえば、温室内気温を調節する目的で、天窗の開閉が行われる。また、土壌含水率の調節を目的として灌水が行われる。

3) 栽培管理は、ノイズを含む場合がある。ノイズは、栽培者の怠慢によるものである。本来ならば、早朝に灌水すべきものを、なんらかの理由で、昼近くになってしまふことがある。また、怠慢によりある操作が不可能な場合、次善の策として他の操作を行うことがある。この場合、コンピュータの学習は非常に困難になることが予想される。このようなノイズと前節で述べたあいまいさは、区別することが困難である場合が多い。

以上より、本研究の目指すシステムが具有すべき学習機能について要約すると次のようになる。すなわち、ノイズを含んだ相互に相関のある条件関数および操作関数の計測値から、あいまいさを含む栽培管理ルール (条件関数の数値の比較の項の AND と OR の任意の組合せからなる条件部と操作関数を結論部に持つルール) を抽出することである。

## 2.3 本報告の範囲

前節で述べた本研究の目指すシステムは Machine Learning の分野での困難な点を含んでいる。それらを次に列挙する。

1) ノイズとあいまいさは、Machine Learning の分野でも、最も難しい課題に属する。あいまいさは、エキスパートシステムなどに取入れられているが、Machine Learning に取入れられた例は少ない。

2) AND と OR の任意の組合せを扱うコンピュータの学習システムは少ない。AND だけ、あるいは OR だけか、あるいは限定された組合せの場合が大半である。

3) 条件関数に派生関数を使用している場合の学習は、きわめて困難となる。条件関数に環境関数、作物関数あるいは時間関数だけを使用する場合は、Michalski (1984) のいう Selective Generalization にあたり、ルールに表われる関数は既知となる。それに対し、派生関数が使われる場合は、Michalski のいう Constructive Generalization に相当し、ルールに表われる関数が未知であるので、その難易度は飛躍的に増大する。なぜなら、派生関数の作成方法は無限にあり、そのどれが該当するかは、あらかじめ分からないからである。多くの場合、Constructive Generalization には、領域固有の知識 (Domain Specific Knowledge) が使われる。

本報告は、本研究の目指すシステムの構築の第一歩であるので、上記の困難さのために、前節で説明した学習機能の多くを省略せざるを得なかった。以下に、本報告の範囲を述べる。

あいまいさとノイズは取り扱わない。また、派生関数も取り扱わないし、AND と OR の組合せにもある制限を設ける (詳しくは別稿, Kurata, 1988 参照)。

### 3. 学習アルゴリズム

開発した学習アルゴリズムをKアルゴリズムと呼ぶ。従来のMachine Learningのアルゴリズムが、なんらかの形の探索を含むのに対し、Kアルゴリズムは条件関数の分布からルールを直接推定するアルゴリズムであり、計算コストが低く、マイクロコンピュータにインプリメントすることも可能である。Kアルゴリズムの詳細は別稿に譲るが、大略次のようである。

Kアルゴリズムは次の2つのプロセスから成り立つ。1つは条件関数の値の分布の解析であり、他の1つは計測例の集合(以後、実例空間とよぶ)を縮小するプロセスである。なお、栽培管理における操作は複数個あるが、以下では1つの操作について考察する。複数個の場合は同じプロセスを繰り返せばよい。

ここで、条件関数の分布の解析で重要となる次の2つの概念を定義する。

P部分集合：条件関数の定義域の部分集合で、操作関数が1となる場合の条件関数の値の分布する部分集合。

N部分集合：条件関数の定義域の部分集合で、操作関数が0となる場合の条件関数の値の分布する部分集合。

ルールの推定はP部分集合とN部分集合の重なり方の解析により行う。すなわち、他の項とOR結合する項の条件関数、他の項とAND結合する項の条件関数と条件部に現われない条件関数の3種の条件関数のP部分集合とN部分集合の重なり方はそれぞれ異なる。したがって、各条件関数のP部分集合とN部分集合の重なり方の解析から、各条件関数が条件部に現われるのか否かが判定できる。また現われる場合、他の項とORで結合されているのか、あるいはANDで結合されているのか、またその条件関数に関する条件は何か(不等号の向きあるいは等号と基準値)も判定可能である。

以上の方法により、条件部が1つの項だけの場合、あるいは、複数の項から成り立つが各項がORあるいはANDの1種類だけで結合されている場合のルールが推定できる。条件部にORとANDの両者が現われる場合は、実例空間の縮小が必要となる。たとえばOR結合のなかにAND結合がある場合や、AND結合のなかにOR結合がある場合である。これらの場合、条件部に現われる条件関数のP部分集合とN部分集合の重なり方は、上述の場合と同じで、その項を推定可能なものもある。しかし、上述の場合とは異なりその項を推定不可能なものもある。そこで、まず推定可能な項を推定し、そこから作ったルール(以後、仮ルールと呼ぶ)によって実例を再

現してみる。仮ルールの形とその再現結果から適当に実例を消去すると、元の実例空間では推定不可能であった項を推定することができるようになる。こうして推定した項と仮ルールの項を結合すれば、元のルールを同定できる。なお、ルールの形式が複雑で以上の方法でも同定不可能な場合もある。しかし、栽培管理ルールではそのような複雑なルールはないと予測できる。

### 4. シミュレーション

#### 4.1 方法

開発したシステムの検証は、本来実際の栽培場面で行うべきものである。しかし、実際の栽培場面に適用するには不可欠な、あいまいさの要素を除去したため、それをあきらめざるをえなかった。そこで、条件関数の架空の時系列をコンピュータ内部で発生し、いろいろな栽培管理ルールを適用した場合のKアルゴリズムの学習機能と問題点を調べた。使用した言語はCである。

操作関数は1つとし灌水を想定した。上述の目的からして1つの操作関数の場合を調べれば十分である。実際の施設園芸を想定し、次の3つの独立関数を作り、他の条件関数はそれらから導いた。ただし、外気温は条件関数には入れなかった。

外気温( $T_{out}$ )：一日の最高温度と最低温度をランダムに与え、最高温度出現時(午後2時)と最低温度出現時(午前5時)の間は、 $\sin$ 関数で結んだ。

日射量( $R$ )：6時から18時まで日射があり、 $\sin$ 関数で変化するものとした。最高値(12時の値)はランダムに与えた。

湿度( $q$ )：各日の6時の値をランダムに与え、その間は直線で補間した。

以上から分かるように、このシミュレーションでは厳密に物理的なモデルは使用していない。湿度は相対湿度とも絶対湿度とも解釈可能である。それは本シミュレーションの目的から、そのような厳密さが要求されないからである。また、同様の理由で、各関数の値は0~100の範囲に入るようにした。単位は無視した。

次に設定した条件関数について述べる。条件関数は計7つとし、上の3つの関数からその時系列を発生した。なお、1日を100等分し、各時刻ごとに条件関数の値を与えた。データは10日間分与えた。各条件関数は次のようである。

環境関数

湿度( $q$ )：独立関数。

日射量( $R$ )：独立関数。

室内気温( $T$ )： $T=0.6R+0.4T_{out}$ で与えた。

土壌含水率( $w$ )：灌水時に100とし、時間ステップご

とに低下するものとした。その低下量は、 $(0.1 - 0.001q)R$ とした。ただし、括弧内は負にはならないものとし、また、 $w$ は10以下にはならないものとした。

作物関数

作物の外観 ( $P$ ):  $P = 0.3w + 0.3T + 0.2R + 0.2q$  で与えた。 $P$ の生理的意味はあえて定義しなかった。

時間関数

$t$  : 一日の時刻(6時からの時間ステップ数)。

$d$  : 前回の灌水日からの日数。

4.2 結果

栽培者のルールをいろいろ想定し、Kアルゴリズムの機能を調べた。試験したルールの数例をTable 1に示す。Table 1には、Kアルゴリズムが学習したルールも示した。両者ともに、ルールの結論部が  $I=1$  となるときの条件部だけを示した。

Table 1に示した例を含め、試験した約20のルールすべてで、栽培管理行為は学習のために用いた時系列で100%再現できた。これはKアルゴリズムが、100%再現できるように実例空間の縮小を行うことによってルールを学習するアルゴリズムであることから当然の帰結といえる。このことは、Kアルゴリズムが栽培管理行為を再現する上で有効なアルゴリズムであることを示している。

しかし、Table 1にみられるようにKアルゴリズムが学習したルールは栽培者のルールと必ずしも同じではない。Table 1の結果を検討すると次のことが分かる。条件部が単項から成り立つような単純なルールの場合、

想定したルールが再現される場合もあるが、多くの場合、学習したルールは想定したルールよりも多くの項をもっている。また、想定したルールの項は学習したルールに含まれている。このことと、学習したルールで栽培管理行為が100%再現できたことから、学習したルールでは表現が冗長になっていることが分かる。これは、条件関数間に相関があるためと考えられる。ある条件関数が想定したルールに現われると、その条件関数と相関のある他の条件関数も学習したルールに現われるのである。大半の相関は、前節で説明した条件関数のモデルによるものと考えられるが、与えたデータに混入した偶然性による相関も考えられる。

実際に適用する場面を想定すると、ルールの冗長な表現をどのように処置するかは難しい。なぜなら、本シミュレーションの場合は、栽培者のルールは分かっているが、実際の場面では分からないからである。本シミュレーションでの冗長な項を除去することは、難しいことではない。除去しても結果の変わらない項を除去すればよいからである。しかし、実際場面の適用では、あいまいさに関与しているから、1つの項の除去が結果になんら影響を与えないとは考え難い。その場合どの程度の変化まで許容するかが問題となろう。したがって、あいまいさを考慮していない現時点では、この問題はこれ以上議論できない。現時点でいえることは次の点である。冗長な表現は、それを制御に使用する限り、条件関数の時系列と各条件関数間の相関関係の確率論的な定常性を前提にするならば問題はない。制御ではなく、人間の理解のための学習であるならば、冗長なルール表現は理解を困難にするから、なんらかの処置が必要であらう。

5. おわりに

施設園芸の栽培管理行為の特徴を示し、Machine Learningの概念を適用した栽培管理の自動化について考察した。また、そのための骨格となるべき学習アルゴリズム(Kアルゴリズム)の可能性と問題点を調べた。その結果、Kアルゴリズムが栽培管理の自動化のためには有効なアルゴリズムであることが分かった。

今後はKアルゴリズムを拡張して、実際に適用可能なものに近づける研究が必要である。多くの残された課題のなかでも、あいまいさを扱うことが特に重要である。また、本報告では計測したデータを同時に全部解析してルールを学習する

Table 1 Examples of if-parts of assumed grower's rules and learned rules by K-algorithm  
The then-part of each rule is  $I=1$ .

| Assumed rules                                     | Learned rules  |
|---|--|
| $t=25$  | $t=25$   |
| $R>80$  | $(R>80) \text{ OR } (T>83)$  |
| $w<40$  | $w<40$   |
| $(R>90) \text{ OR } (30<t<35)$                    | $(R>90) \text{ OR } (30<t<35) \text{ OR } (T>83) \text{ OR } (P>83)$   |
| $(q<50) \text{ AND } (d=2)$                       | $(19<q<50) \text{ AND } (d=2) \text{ AND } (16<T<46) \text{ AND } (R<27) \text{ AND } (w<22) \text{ AND } (P<32) \text{ AND } (t<42)$  |
| $(R>70) \text{ OR } ((w<50) \text{ AND } (P<50))$ | $(R>70) \text{ OR } (T>80) \text{ OR } (w<18) \text{ OR } (P>78) \text{ OR } (P<26) \text{ OR } ((28<w<50) \text{ AND } (34<P<50) \text{ AND } (30<T<63) \text{ AND } (18<q<67) \text{ AND } (25<R<62) \text{ AND } (7<t<40))$ |

方法を扱ったが、実際場面での適用を考えると、実時間で計測しつつ学習するいわゆる incremental learning の要素が必要であろう。しかし、incremental learning はノイズに弱いという欠点をもつ。これらの点の解決は今後の課題である。

謝 辞

本研究の一部は文部省科学研究補助金（課題番号 61560286）による。

引用文献

Kurata, K., 1988: Greenhouse control by machine learning. *Acta Horticulturae*, **230**, 195-200.

蔵田憲次・大原源二, 1988: 栽培管理と作物の認知に関する考察, *農業気象*, **44**, 33-36.

日本施設園芸協会, 1986: 施設園芸におけるマイクロコンピュータ導入の手引, p.131.

Michalski, R. S., 1984: A theory and methodology of inductive learning. In *Machine learning an artificial intelligence approach* (ed. by R. S. Michalski et al.). Springer-Verlag, Berlin, 83-134.

Winston, P. H., 1975: Learning structural descriptions from examples. In *The psychology of computer vision* (ed. by P. H. Winston). MacGraw-Hill, 白井良明, 杉原厚吉訳, 1979: コンピュータービジョンの心理, 産業図書, 173-235.

Summary

Computers have been applied to crop management in protected cultivation and have proved to be of great use. However, in the present computer systems, control algorithm is fixed and users can only change the setpoints of the controlled variables. Crop management is strongly influenced by the local conditions of climate, soil, etc. and well trained and eager growers manage their crops in the way which is suitable under the given local conditions and is obtained through the long period experience.

This study aims at developing a system which learns grower's managing methods (rules, hereafter) through measuring environmental factors, crop status, if possible, and grower's behavior. After learning the grower's rules, the learned rules will be applied to the automatic management. Thus, the grower will be released from the management labor without losing their own personal preference and principles in crop management. This system can also be applied to the rule acquisition and analysis of well trained growers. This report presents considerations on the characteristics of the growers' rules and the possibilities and problems of the simple learning algorithm developed (K-algorithm, hereafter), which will be the framework of the system we aim at. Details of K-algorithm are given in another report (Kurata, 1988).

Considerations on the grower's rules revealed that there are many difficult aspects for the computer in learning, for examples, the problem of fuzziness associated with the human feelings, the problem of noise and how to find out a function which the grower's rule has but not directly measurable. All these points are omitted in the present study and left for the future study.

A simple simulation revealed that K-algorithm is very powerful in imitating the grower's behavior but the learned rules are somewhat longer than the assumed grower's rules, examples of which are listed Table 1. This redundant expression of the learned rule is due to some correlations among environmental factors, crops status and time. If we assume stochastic stationary state of these functions and their correlations, this brings about no problem in applying the learned rules to the automatic management.