

姿勢角センサを用いた車両姿勢の動的計測法

岩岡正博*・仁多見俊夫*・小林洋司*・南方康*

Methods of Measuring a Vehicle's Leaning with a Posture Sensor

Masahiro IWAOKA*, Toshio NITAMI*, Hiroshi KOBAYASHI* and Yasushi MINAMIKATA*

I. はじめに

近年, トラクタ, 林内作業車などの車両系機械が多く作られるようになっており, これらを使用した林業作業も増化している。また, 最近導入されつつある高性能多工程処理機械の多くは, エクスカベータやブルドーザなどの車両系機械をベースマシンとして, 直接林内へ進入して作業を行っている。

機械を使用して林内の各種作業を行う場合, 作業地の傾斜を克服することが重要な課題であるが, この問題を解決するための方策としては, 車体上部の運転席を必要に応じて傾け, 姿勢制御のできる機構を持たせることによって重心を移動し, 傾斜地における安定性, 走破性を向上させた機械や, 5脚の歩行機械などが実用化されている。しかしながらこれらのベースマシンは, 操作が非常に複雑になるため, 迅速な作業は行いえない。したがって, これらの操作および制御を, 可能な限り自動化することが望まれる。現在, 林業用車両の姿勢を自動的に制御する研究としては, 走行装置を上下に動かすことによって, 斜面上で車体を水平に保つ段軸式トラクタの姿勢を, コンピュータを用いて適切に自動制御する研究¹⁾がみられる程度である。

車両の姿勢をコンピュータなどを用いて自動的に制御するには, まず車両の姿勢を動的にかつ自動的に計測することが必要である。従来筆者らは, ジャイロなどを用いて車両の挙動の計測を行う方法について研究を行ってきた^{2,3)}。しかし, ここで使用した回転式ジャイロは, 検出精度などは十分であったが, きわめて高価な上, 振動などに弱いため, 制御用に一般的に使用するには適さなかった。この経験から, 車両姿勢の自動制御に用いる計測器の条件として, 以下に示す条件を満足することが望ましいと考えている。

1. リアルタイムで計測できること。
2. コンピュータに直接入力可能な形式のデータを出力すること。
3. 車両に搭載可能な程度の大きさであること。
4. 制御に必要な精度を有すること。
5. 不整地を走行する車両に搭載可能な程度の耐震性を持つこと。

このたび, 上記の条件を満たす, 比較的安価で構造の簡単な姿勢角センサ (T 社製 TA7175N 1) を用いて, 車両の姿勢角の動的計測システムを作成し, 車両姿勢の自動制御などへの適応性を検討したので, ここに報告する。

* 東京大学農学部林学科
Department of Forestry, Faculty of Agriculture, The University of Tokyo.

表-1 姿勢角センサの仕様
Table 1. Specifications of the posture sensor

項目	仕様値	備考
立ち上がり時間	5分以下	
検出範囲		
ピッチ角、ロール角	±45°	
角速度	±100°/s	
検出精度		
ピッチ角、ロール角	±1°以下 (検出範囲 15°未満) ±2°以下 (検出範囲 15°以上)	使用時間: 1時間
角速度	±2%F.S 以下	スケールファクタの非直線性
出力データ分解能		
姿勢角	0.1°/LSB	12 bit 出力
角度	0.1°/s/LSB	12 bit 出力
動特性	25 Hz 以上	-3 db ポイント
電源	+11～+18 VDC	
消費電流	1.3 A 以下	
外形寸法	W×D×H 130×100×110 mm	
重量	1.7 kg	

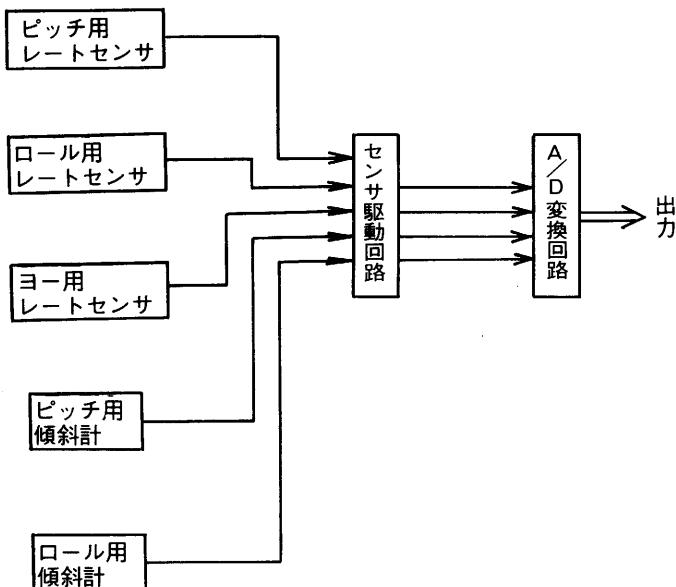


図-1 姿勢角センサの内部構成
Fig. 1. Structure of the posture sensor.

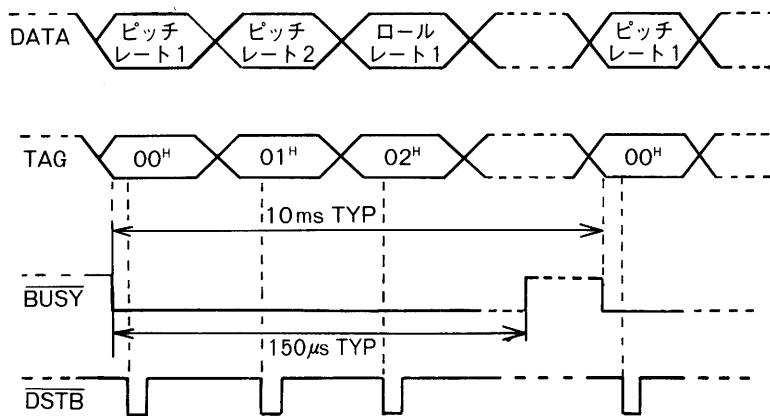


図-2 姿勢角センサのデータ出力形式
Fig. 2. Data forms of the posture sensor.

II. 車両姿勢角の動的計測システム

1. 計測装置の概要

本研究で用いた姿勢角センサは、表-1に示すとおり小型軽量なケースの中に、3台のレートセンサと2台の傾斜計が組み込んでおり(図-1)、ロール角、ピッチ角、ロールレート、ピッチレート、ヨーレートの各データをデジタル信号として出力する。このため、パラレルI/Oインターフェースを介することによって、直接コンピュータに入力することが可能である。出力するデータの分解能は0.1度、検出精度は±1度であり、車両の姿勢制御に用いるデータとしては、十分な精度を有すると考えられる。また、応答性は25Hz以上であり、かつデータ出力のインターバルは10m秒であるので、リアルタイムの計測にも十分使用可能である。

この姿勢角センサからは、図-2に示すとおり、8ビット(1バイト)のデータと、4ビットのタグ、および各1ビットのビジー信号とデータストローブ信号が同時に outputされる。データはそれぞれ2バイト使用しており、このうちの12ビットで表されている。また、その最上位ビットは、符号ビットとなっている。データの出力順序は、ピッチレート、ロールレート、ヨーレート、ピッチ角、ロール角、非データ、ステータスとなっており、それぞれ上位バイト、下位バイトの順に出力される。したがって、出力データの1組みは14バイトとなる。また同時に、タグによってその時出力されているデータの種別を示すとともに、ビジー信号が1組みのデータの先頭を示し、データストローブ信号が各1バイトの出力タイミングを示している。

このデータ出力のタイミングを実測した結果、1組14バイトのデータを出力するのに要する時間は、約260μ秒であった。したがって、データの出力速度は、約18μ秒/バイト(約5,300バイト/秒)と、きわめて高速であった。

パラレルI/Oインターフェースとしては、M社製DIO-3298CPCを使用した。この入力レベルはTTL(5V)で、入力ポートは24ビット1ポート、または8ビット3ポートとして使用可能である。ここでは、8ビット3ポートとして使用して、データを1つのポートに割り当て、タグお

表-2 パラレル I/O インターフェースの仕様
Table 2. Specifications of the parallel I/O interface

項 目	仕 様
入力ポート	24 ビット・TTL レベル (8 bit × 3 ポート / 10 k でプルアップ)
I/O アドレス	上位 8 bit および中位 4 bit を DIP スイッチで設定 下位 4 bit は固定
電 源	+5V, 0.35 A (パソコン側より供給)
使用温度範囲	0~45 度 C

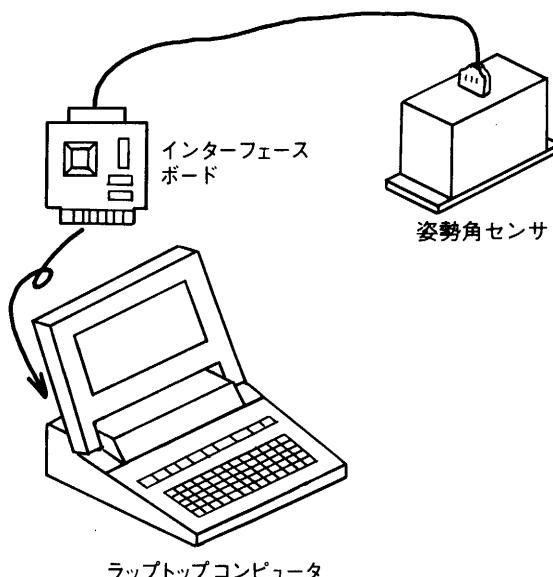


図-3 計測装置の関係
Fig. 3. Relationship of the sensors.

およびビジー信号、データストローブ信号をまとめて別の 1 つのポートに割り当てた。このインターフェースの仕様を表-2 に示す。

データの入力および保存には、16 ビットラップトップコンピュータ (E 社製 PC286LE) を使用した。これは可搬式になっており、底面積が 345×335 mm と小さいので、計測システム全てを計測車両に搭載して、計測することが可能である。

以上の計測器の関係を図-3 に示す。

2. 計測用プログラムの仕様

計測用のプログラムは、C 言語を使用して開発した。図-2 に基づいて計測用プログラムの流れを示すと、

1. ビジー信号によって、1 組みのデータの先頭を待つ。
2. データストローブ信号を確認して、各データを 1 バイトずつ読み込む。

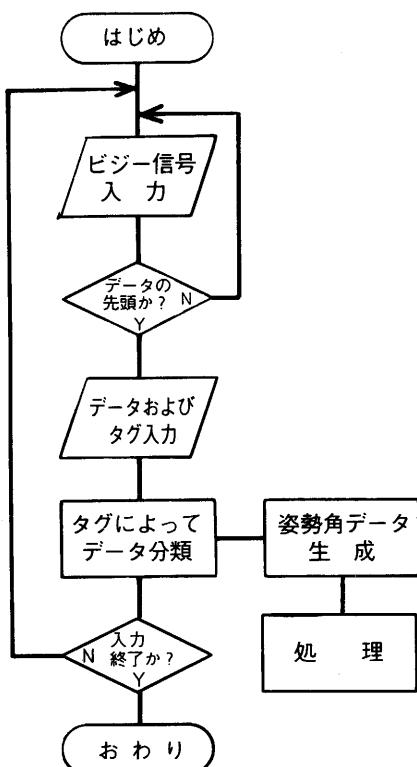


図-4 計測用プログラム流れ図

Fig. 4. program flow chart.

を生成する。

5. データに対応した処理を行う。

このように、データ入力部分とデータ分類部分を分離し、データを一度メモリー中に保存してから、再度読み出すようにした。データを分類するに当たっては、データ、タグともに負論理で出力されているため、ビット反転する必要があった。ビット反転して得られたタグに基づいて、データを分類した後、上位バイトを8ビットシフトして加えることによって、データの上下位バイトを結合した。また、データを格納する変数が16ビットの符号付き整数であるのに対して、データは12ビットの符号付き整数であるので、16進7ff以上のデータからは、16進1000を減じて、負の数とした。

得られたデータに対して、適当な処理を行った後、再度データを読み込むことをくりかえすが、ここで、スリープタイマールーチンを加えることによって、一定間隔でデータを入力することが可能である。プログラム例を、付表として添付する。

3. データの上下位バイトを結合して、姿勢角データを生成する。

4. データに対応した処理を行う。

となる。しかし、前章で示したとおり、姿勢角センサのデータ出力速度はきわめて高速なため、データストローブ信号を確認してからデータを読み込むためには、最低でも10,600バイト/秒以上の入力速度が要求される。このことから、ここで使用したラップトップコンピュータの動作速度では、データストローブ信号を確認していると、必要な速度が得られないことが明らかになった。また、読み込みと同時にデータの上下位バイトの結合を行った場合も、必要な速度が得られなかつた。さらに、読み込み動作を単純ループとしてまとめた場合もまた、必要な速度は得られなかつた。

この結果、プログラムの流れは、以下のようになつた(図-4)。

1. ビジー信号によって、1組みのデータの頭を待つ。

2. データとタグを、データ出力の終りまで一度に読み込む。

3. タグによってデータを分類する。

4. 上位バイトを結合して、姿勢角データ

表-3 計測実験において設定した各試験路の特徴

Table 3. Characteristics of the test courses

	横 断 傾 斜(度)				地表状態
	最 大	最 小	平 均		
走行路 a	2.7	0.5	1.6	極めて硬	
走行路 b	6.4	4.0	5.3	極めて硬	
走行路 c	13.9	6.2	10.0	軟	
走行路 d	22.9	8.5	13.5	軟	

III. 計測システムを用いた計測例と検討

1. 計測方法

ここで作成した計測システムを用いて、宇都宮大学船生演習林において小型林内作業車による姿勢角の計測実験を行った。実験は、斜面上を等高線方向に走行する車両の、側方姿勢角（ロール角）を計測することを目的として、横断傾斜の異なる長さ 20 m の直線走行試験路を、等高線方向に 4 本設定して行った。各試験路の横断傾斜と地表状態は表-3 に示すとおりであるが、最大傾斜は 22.9 度で、これ以上の傾斜地は転倒の危険があるため走行できなかった。実験に用いた車両は、硬式懸架装置を持つキャタトラと、軟式懸架装置を持つリョウシン号である。計測用プログラムは、0.1 秒ごとにデータを読み込んでメモリーに保存し、計測終了後に姿勢角データに変換して、フロッピーディスクへ書き込むように作成した。

実験はまず、実験車両を用いて試験路上を数回走行して、試験路の表面を踏み固めた後、試験路上の 4 点において、各車両の静止時の姿勢角を計測した。その後、車両を静止させた点の横断傾斜を、レベルを用いて測量した。こうして準備した後に、車両を走行させて、走行時の車両の姿勢角を計測するとともに、先に静止させた地点の通過時刻を記録して、地表傾斜と比較できるようにした。前進、後退一往復の計測を行った後、地表傾斜を再度測量し、車両の進行方向を変えてさらに一往復の計測を行い、最後にもう一度測量を行った。以上の手順を、各車両について行った。

走行には、通常搬出作業を行う場合に使用するギヤを用いたため、各車両の平均走行速度は、キャタトラ 0.34 m/s (0.28~0.46/s)、リョウシン号 0.38 m/s (0.33~0.44 m/s) と、きわめて低速であった。

2. 計測結果と検討

計測結果を、図-5, 6 に示す。図-5 はキャタトラを用いて、試験路 c を走行した時の車体の側方姿勢角であり、図-6 はリョウシン号を用いて試験路 b を走行した時の車体の側方姿勢角である。図中、姿勢角が正のときは車体が右に傾いていることを、また負のときは車体が左に傾いていることを、それぞれ表している。また、図中の▲は走行点の横断傾斜を示している。両図とともに、1 度以下の細かい車体の振動まで示しており、車両の姿勢角を忠実に再現していると考えられる。またこれを地表傾斜と比較することによって、車両が地表の傾斜に沿って走行していく様子が読み取れる。以上から、本計測システムは、走行する車両に搭載した状態で、その姿勢角をリ

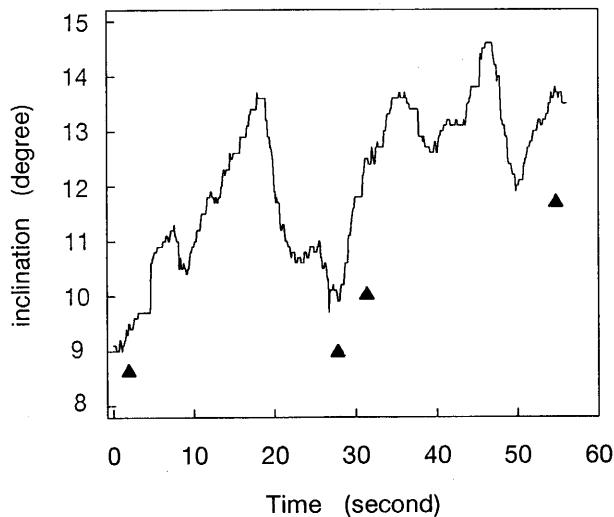


図-5 キャタトラ走行時の側方姿勢角
▲, 地表傾斜

Fig. 5. Rolling angles of the "Catatora" while in motion.
Legend: ▲, Terrain inclination.

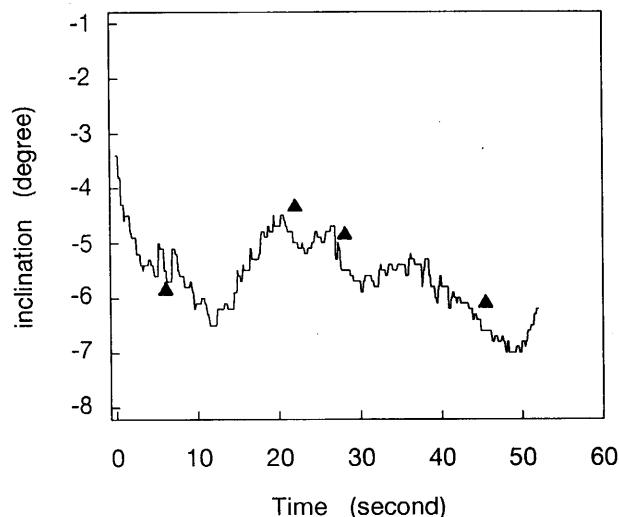


図-6 リョウシン号走行時の側方姿勢角
▲, 地表傾斜

Fig. 6. Rolling angles of the "Ryousin-go" while in motion.
Legend: ▲, Terrain inclination.

アルタイムに計測しうるといえよう。

以上の計測結果を用いて、車両の側方姿勢角と地表傾斜との関係を検討した⁴⁾。図-7に地表傾斜とキャタトラの側方姿勢角との関係を、図-8に地表傾斜とリョウシン号の側方姿勢角との関

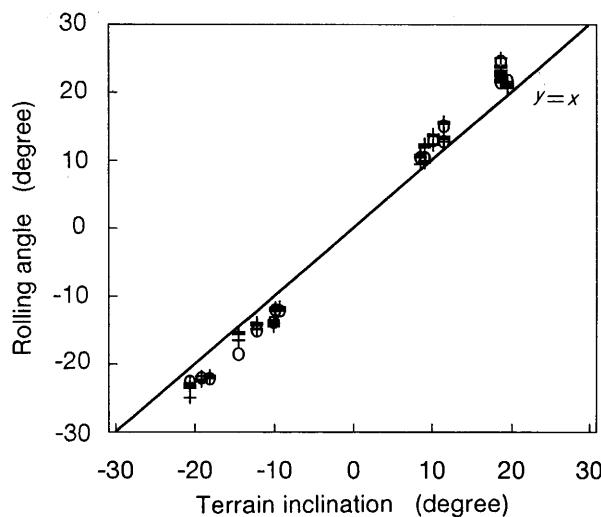


図-7 地表傾斜とキャタトラの側方姿勢角との関係
○, 静止時; +, 走行時

Fig. 7. Relationships between terrain inclinations and rolling angles of the "catatora".
Legend: ○, Motionless; +, Moving.

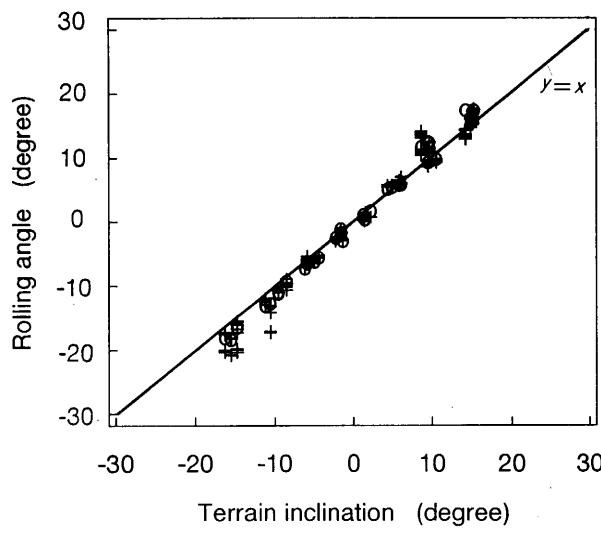


図-8 地表傾斜とリョウシン号の側方姿勢角との関係
○, 静止時; +, 走行時

Fig. 8. Relationships between terrain inclinations and rolling angles of the "Ryousin-go".
Legend: ○, Motionless; +, Moving.

係を示す。図中、○は静止時の側方姿勢角を、+は走行時の側方姿勢角をそれぞれ示し、直線は姿勢角が地表傾斜に等しい場合の値を示す。両図から、側方姿勢角は地表傾斜と直線関係にあることが分かる。また、地表傾斜よりもいくぶん大きく傾いていることが分かる。ここで、両者を

表-4 地表傾斜と車両の側方姿勢角との一次回帰結果

Table 4. Regression coefficients and coefficients of correlation between terrain inclinations and rolling angles

	回 帰 係 数	相 関 係 数
キ タ ラ 静 止	1.2	0.998
キ タ ラ 走 行	1.2	0.998
リ ヨ ウ シン 号 静 止	1.1	0.997
リ ヨ ウ シン 号 走 行	1.1	0.992

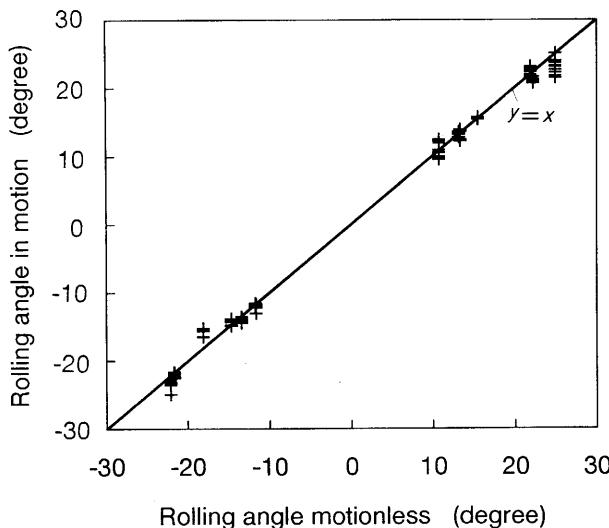


図-9 キャタトラの側方姿勢角の静止時と走行時の変化

Fig. 9. Relationships between the "Catatora's" rolling angles while motionless and in motion.

一次回帰した結果の回帰係数と相関係数を表-4に示すが、相関係数はいずれも極めて1に近く、側方姿勢角と地表傾斜との相関性が非常に高いことを示している。また、回帰係数は1.1～1.2であり、側方姿勢角は地表傾斜より1～2割も大きくなっていることを示している。この原因として、軟式懸架装置を持つリヨウシン号の場合は、懸架装置の動きによって地表傾斜よりも大きく傾くことが考えられる。しかし、硬式懸架装置を持つキャタトラの場合は、懸架装置の動きによって姿勢角が変化することはありえないで、走行路の沈下によって地表傾斜が変化し、車両の姿勢角に変化が現れたと考えられる。すなわち、試験路の表面は、計測前によく踏み固めたにもかかわらず、車両通過時だけ沈下し、通過後に復元するという弾力性を持っていたと考えられる。この結果、車両走行後に測量した試験路の傾斜と比べても、車両の側方姿勢角は大きくなったのであろう。

次に、車両の側方姿勢角が、静止時と走行時とでどのように異なるか検討した。キャタトラの静止時の側方姿勢角と走行時の側方姿勢角との関係を図-9に、リヨウシン号の静止時の側方姿勢角と走行時の側方姿勢角との関係を図-10に示す。図中の直線は、静止時と走行時の側方姿勢角が等しいときの値を示す。両図とも、プロットした点は直線上に位置しており、車両走行時の

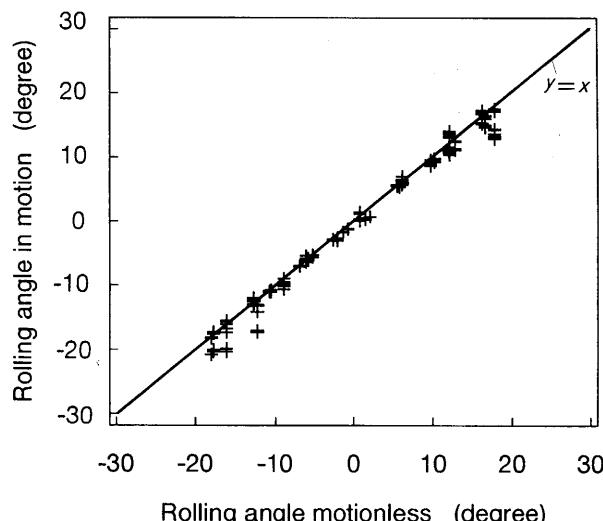


図-10 リョウシン号の側方姿勢角の静止時と走行時の変化

Fig. 10. Relationships between the "Ryousin-go's" rolling angles while motionless and in motion.

表-5 静止状態と走行状態における車両の側方姿勢角の一次回帰結果

Table 5. Regression coefficients and coefficients of correlation between rolling angles while motionless and moving

	回 帰 係 数	相 関 係 数
キ キ ャ タ ト ラ	0.99	0.998
リ ョ ウ シ ン 号	1.0	0.994

側方姿勢角は静止時のそれとほぼ等しいことが分かる。両者を一次回帰した結果の回帰係数と相関係数を表-5に示すが、いずれの値も極めて1に近く、車両走行時の側方姿勢角と静止時の側方姿勢角がほぼ等しいことを裏付けている。ここで、硬式懸架装置を持つキャタトラの場合、車両の姿勢角は静止時、走行時とも等しいと仮定すると、本計測システムの検出誤差は1%となり、必要十分な精度を有しているといえる。したがって、車両の姿勢制御に十分使用可能であると考えられる。また上記の結果から、リョウシン号のように軟式懸架装置を持つ車両であっても、本実験のように走行速度が極めて低速な場合には、走行することによって生じる振動などの影響で、車両の姿勢角が静止時から大きく変化することはないと考えられるため、車両が低速で走行する場合の安定性を検討するために、静止転倒角を用いることは妥当であるといえよう。

IV. お わ り に

本研究で作成した車両姿勢角計測システムで使用したセンサは、25 Hz以上の応答性と±1度の精度を兼ね備えており、かつ小型軽量であるため、目標としていた車両の姿勢制御に用いるために必要な条件を満たしていた。この計測システムの車両姿勢制御への適応性を検討するため、傾斜地を走行する車両に搭載して、姿勢角を計測する実験を行った結果、リアルタイムで、かつ

十分な精度を有する出力が得られた。このことから、今後必要とされる、車両姿勢の自動制御に、十分応用可能であるとの見通しが得られた。また、この計測システムを用いて行った計測試験の結果から、斜面上を低速で走行する車両の側方姿勢角は、静止時の側方姿勢角にはほぼ等しいことが明らかになった。今後は、車両系機械や、脚式歩行機械の姿勢制御に、本計測システムを応用していくことを考えている。この制御技術の発展により、機械化作業による将来の林業作業の安全性、快適性に寄与しうることが望まれる。

なお、計測実験を行うに当たってご支援いただいた、宇都宮大学および附属演習林の教官ならびに職員の方々に、心から御礼申し上げる。

本研究は、平成元年度文部省科学研究費補助金（一般研究（B）63480060）を得て行い、一部を第101回日本林学会大会において発表した。

要 旨

傾斜不整地における車両系機械の安定性を向上させる手段の1つとして、姿勢を適切に制御することが考えられるが、この制御を自動的に行うためには、姿勢角を正確にかつリアルタイムで計測することが必要である。そこで、ピッチ角とロール角の2軸廻りの傾斜角と、ピッチレート、ロールレート、ヨーレートの3軸廻りの角速度を同時に検出し、離散データとして出力する姿勢角センサなどを用いて、車両走行時の姿勢角を動的に計測するシステムを作成した。このシステムは、必要な精度と応答性を兼ね備えており、また小型軽量な検出部であるため、傾斜地を走行する車両に搭載して、その姿勢角をリアルタイムに、かつ十分な精度で計測することができた。また、この計測システムを用いて、斜面上を走行する車両の側方姿勢角を計測する実験を行った結果、低速で走行する車両の側方姿勢角は、静止時の側方姿勢角にはほぼ等しく、地表傾斜を示すことが明らかになった。

キーワード： 姿勢角、傾斜地、動的計測システム、姿勢制御

引 用 文 献

- 1) 福田章史・市原恒一・朝日一司・毛綱昌広：林業用段軸車両の姿勢制御。日林誌 72, 239-246, 1990.
- 2) 岩岡正博・仁多見俊夫・南方 康：車両走行時の挙動計測に関する研究。39回目林関東支論, 251-252, 1987.
- 3) ———・———・小林洋司・南方 康：不整地走行車両の挙動計測法に関する研究。100回目林論, 749-752, 1989.
- 4) ———・小林洋司・仁多見俊夫・南方 康：不整地を走行する車両の挙動に関する研究—斜面上における車両の側方の傾きと地表傾斜との関係—。101回目林論, 669-670, 1990.

(1990年10月31日受理)

Summary

To improve the stability of the vehicles in an inclined position off-road, it is one of these methods that best controls its posture. To control it automatically, it is necessary to measure its posture angles accurately and in real-time. With a posture sensor that monitors two angles (pitching angle and rolling angle) and three angular velocities (pitching rate, rolling rate, and yawing rate) simultaneously and records them as digital data, the system, which measures posture angles dynamically at the time of a vehicle's movement, is

complete. This system combines the necessary precision and responsiveness, and its detecting element is small and lightweight. Therefore, it can be mounted easily on a vehicle that runs on inclined ground, and it can measure posture angles with sufficient precision and in real-time. Experiments that measured the rolling angles of vehicles running on slopes, were made with this instrumentation system. Consequently, it became clear that the rolling angles of the vehicles running at slow speeds were almost equal to their rolling angles when they were motionless on same terrain, and they indicated terrain inclination.

Key words: Posture angle, Inclined ground, Dynamic instrumentation system, Posture control

付表-1

```

/* sample.c
 * TA7175N1 計測用プログラムサンプル
 * © Masahiro Iwaoka 1990.10
 * with Microsoft C Compiler (cl -Ox)
 * 注) これは、データを読むだけで、"何もしない" プログラムです */
#include <stdio.h>
#include <conio.h>

#define PORT      (unsigned)0x40d0      /* データポートアドレス
                                         (設定により変わる) */
#define TPORT     (unsigned)0x40d2      /* タグポートアドレス */
#define BUFF      34

#define data_read(a) data[(a)][0]=inp(TPORT);data[(a)][1]=inp(PORT)
#define hanten(a)  (^data[(a)][1])&0xff

static unsigned char   data[BUFF][2];    /* データの一時格納用領域 */
struct angles {
    int pitch_rate; /* pitching rate */
    int roll_rate;  /* rolling rate */
    int yaw_rate;   /* yawing rate */
    int pitch;       /* pitching angle */
    int roll;        /* rolling angle */
    int yaw;         /* yawing angle */
    int state;       /* status */
};

main()
{
    int i;
    struct angles odata;
    void makedata();

    fprintf (stderr, "「リターン」キーでスタート");
    while (getchar () != '\n');
    fprintf (stderr, "HOMEキーで stop");

    for (i = 0; i = 0){
        while (((inp (TPORT)) & 0x10) == 0x00);
        while ((inp (TPORT)) & 0x10);
}

```

Continued.

```
/* 以下は冗長であるが、必要な速度を得るために仕方ない */
data_read(i);          /* 1 */
data_read(i+1);         /* 2 */
data_read(i+2);         /* 3 */
data_read(i+3);         /* 4 */
data_read(i+4);         /* 5 */
data_read(i+5);         /* 6 */
data_read(i+6);         /* 7 */
data_read(i+7);         /* 8 */
data_read(i+8);         /* 9 */
data_read(i+9);         /* 10 */
data_read(i+10);        /* 11 */
data_read(i+11);        /* 12 */
data_read(i+12);        /* 13 */
data_read(i+13);        /* 14 */
data_read(i+14);        /* 15 */
data_read(i+15);        /* 16 */
data_read(i+16);        /* 17 */
data_read(i+17);        /* 18 */
data_read(i+18);        /* 19 */
data_read(i+19);        /* 20 */
data_read(i+20);        /* 21 */
data_read(i+21);        /* 22 */
data_read(i+22);        /* 23 */
data_read(i+23);        /* 24 */
data_read(i+24);        /* 25 */
data_read(i+25);        /* 26 */
data_read(i+26);        /* 27 */
data_read(i+27);        /* 28 */
data_read(i+28);        /* 29 */
data_read(i+29);        /* 30 */
data_read(i+30);        /* 31 */
data_read(i+31);        /* 32 */
data_read(i+32);        /* 33 */
data_read(i+33);        /* 34 */

makedata( &odata );
action ( &odata );
if (kbhit()){
    if (getch() == 0x1a){
        exit (0);
    }
}
}
```

Continued.

```

/* タグによってデータを分類 */
void
makedata( angle )
struct angles *angle;
{
    unsigned char tag1, tag2 = 0x0d; /* タグ */
    unsigned char prh, prl, rrh, rrl, yrh, yrl, ph, pl, rh, rl,
                  yh, yl, sth, stl; /* 各データの上下位バイト */
    int i;

    for (i = 0; i < BUFF; i++){
        tag1 = (~(data[i][0])) & 0x0f;
        if (tag1 != tag2){
            if (tag1 < tag2 && tag2 != 0x0d){
                continue;
            }
            switch (tag1){
                case 0x00 :
                    prh = hanten(i);
                    break;
                case 0x01 :
                    prl = hanten(i);
                    break;
                case 0x02 :
                    rrh = hanten(i);
                    break;
                case 0x03 :
                    rrl = hanten(i);
                    break;
                case 0x04 :
                    yrh = hanten(i);
                    break;
                case 0x05 :
                    yrl = hanten(i);
                    break;
                case 0x06 :
                    ph = hanten(i);
                    break;
                case 0x07 :
                    pl = hanten(i);
                    break;
                case 0x08 :
                    rh = hanten(i);
                    break;
            }
        }
    }
}

```

```

        Continued.

case 0x09 :
    rl = hanten(i);
    break;
case 0x0a :
    yh = hanten(i);
    break;
case 0x0b :
    yl = hanten(i);
    break;
case 0x0c :
    sth = hanten(i);
    break;
case 0x0d :
    stl = hanten(i);
    break;
default :
    break;
}
}
tag2 = tag1;
}

angle -> pitch_rate = cat_high_low (prh, prl);
angle -> roll_rate = cat_high_low (rrh, rrl);
angle -> yaw_rate = cat_high_low (yrh, yrl);
angle -> pitch = cat_high_low (ph, pl);
angle -> roll = cat_high_low (rh, rl);
angle -> yaw = cat_high_low (yh, yl);
angle -> state = cat_high_low (sth, stl);
}

/* データの上下位バイトを結合 */
int
cat_high_low(high, low)
unsigned char  high, low;
{
    int adata;

    adata = ((high << 8) + low) & 0xffff;
    if (adata > 0x07ff) adata -= 0x1000;
    return (adata);
}

/* アクション */
int

```

Continued.

```
action ( datas )
struct angles *datas;
{
    return(1);
}
```