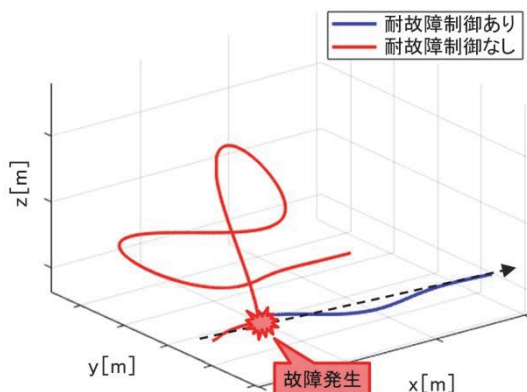


アクチュエータの冗長性を利用した 無人ビークルの耐故障制御機能の開発

Development of Fault-Tolerant Control for Unmanned Vehicles
Using Redundancy Actuators



小田 一道*1
Kazumichi Oda

安達 丈泰*2
Takehiro Adachi

粟屋 伊智郎*3
Ichiro Awaya

無人ビークルには、アクチュエータ故障が発生したとしても、自律的にミッションを継続できるレジリエンス性能が求められている。そこで本報では、レジリエンス性能を実現するために、無人ビークルに冗長性を持つように配置されているアクチュエータを利用した耐故障制御を提案する。具体的には、アクチュエータ毎に推定した舵角推定値を比較することでアクチュエータ故障を検知する故障検知器と、検知したアクチュエータ故障の情報から故障の影響を取り除く故障の非干渉制御器を構築することで、無人ビークルにアクチュエータ故障が発生した場合にも移動を継続できる耐故障制御機能を開発した。さらに、開発した耐故障制御の有効性を検証するため、シミュレーション評価を行った。

1. はじめに

近年、省人化や危険な環境での作業の代替等を背景に、無人搬送車 (Automated Guided Vehicle; AGV)、無人航空機 (Unmanned Aerial Vehicle; UAV)、自律型無人潜水機 (Autonomous Underwater Vehicle; AUV) などの無人ビークルの開発が数多く進められている。これら無人ビークルは、人の介在しない場所で活動する特徴から、アクチュエータに故障が発生した場合であっても、自律的に移動を継続できるレジリエンス性能が求められている⁽¹⁾。

レジリエンス性能を実現する手法の一つとして、あるアクチュエータが故障した場合にも他のアクチュエータを用いて制御が可能のように、一つの運動に対して複数のアクチュエータが影響するように配置された、冗長性を持つアクチュエータを利用した耐故障制御が考えられる。例えば、4つの冗長性を有した舵を備える AUV“ゆめいるか”では、舵を正常として設計したロバスト制御により、一部の舵が故障した際にも制御性能は劣化するがビークル運用に必要な最低限の運動制御性能が確保可能であることが報告されている⁽²⁾。また、UAV においては、故障の影響を事前に考慮したロバストモデル予測制御により、プロペラモータの一部が故障した場合でも、安定性を損なわずにホバリング可能としている⁽³⁾。

一方で、アクチュエータの故障を検知し、積極的に故障対処に用いる方法についても検討されている。例えば、故障発生の前後でアクチュエータから出力される操作量が等しくなるように、連立方程式を解くことで入力量の分配量を計算するコントロールミキサー法^{(4)~(6)}や、事前に設定した故障モードに対して制御器を設計し、故障発生時に該当する制御器に切り替える手法⁽⁷⁾が提案されている。しかし、例えばモータと接続されているリンク機構が外力により固着するなど、センサ

*1 デジタルイノベーション本部 CIS 部

*2 デジタルイノベーション本部 CIS 部 首席技師

*3 デジタルイノベーション本部 CIS 部 ME 工博

からは検知できないような故障が発生した場合には、制御対象からセンシングした信号を用いて故障を検知する故障検知器を別途構築する必要がある。

そこで本報では、無人ビークルに配置されているアクチュエータの冗長性を利用した耐故障制御を紹介する。具体的には、アクチュエータ毎に状態を推定し、その舵角推定値を比較することでアクチュエータ故障を検知する故障検知器と、検知したアクチュエータ故障の情報から故障の影響を取り除く耐故障制御器を構築することで、無人ビークルにアクチュエータ故障が発生した場合にも、移動を継続できる耐故障制御機能を開発した⁽⁸⁾。

2. アクチュエータの冗長性を有する無人ビークル

本報で制御対象とする無人ビークルの一例として、4つの舵を備えるAUVを図1、図2に示す。ビークルは機体座標系の x, y, z の3軸まわりの回転運動である姿勢運動(ロール, ピッチ, ヨー)を行いながら前進することで、 x, y, z 方向に移動する。例えば、深度方向に上昇する場合には、ピッチ角を上方向につけて前進することで上昇する。ある平衡点における、姿勢運動の連立微分方程式を式(1)に示す。

$$\frac{d}{dt} \begin{bmatrix} \Delta p \\ \Delta q \\ \Delta r \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} \Delta p \\ \Delta q \\ \Delta r \end{bmatrix} + \begin{bmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \\ e_{31} & e_{32} \end{bmatrix} \begin{bmatrix} \Delta \phi \\ \Delta \theta \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \end{bmatrix} \begin{bmatrix} \Delta \delta r_1 \\ \Delta \delta r_2 \\ \Delta \delta r_3 \\ \Delta \delta r_4 \end{bmatrix} \quad (1)$$

ここで、 a_{ij}, b_{ij}, e_{ij} は運動方程式の係数であり、 Δ は平衡点からの変化量を表す。また、 θ, ϕ は図1,図2に示す姿勢角、 p, q, r は姿勢角速度、 δr_i は制御器によって操作される舵# i の舵角を表す。

本研究で制御対象とする無人ビークルは、アクチュエータに冗長性を有しているため、3軸の姿勢運動に用いることができるアクチュエータの数が4以上となる。

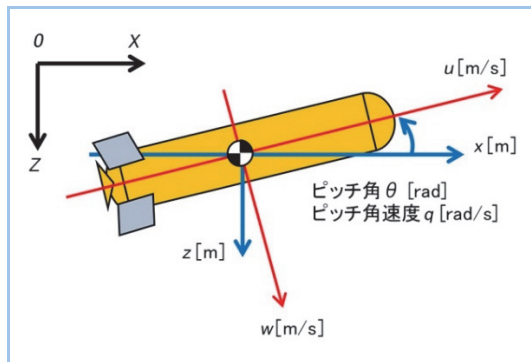


図1 無人ビークル(AUV)のイメージ
(X-Z 平面図)

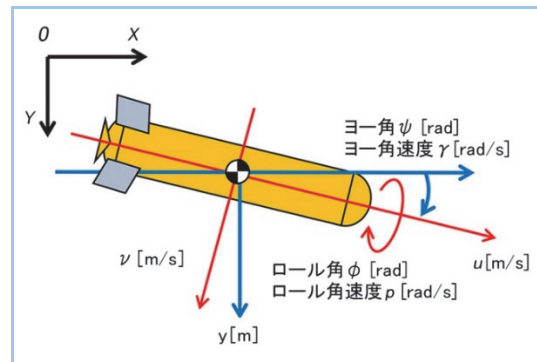


図2 無人ビークル(AUV)のイメージ
(X-Y 平面図)

3. 耐故障制御の開発

図3に開発した耐故障制御機能のブロック線図を示す。耐故障制御機能は、アクチュエータの故障箇所と故障の大きさを推定する故障検知器と、故障検知器から出力される故障状態量を用いて故障の影響を打ち消す耐故障制御器で構成される。

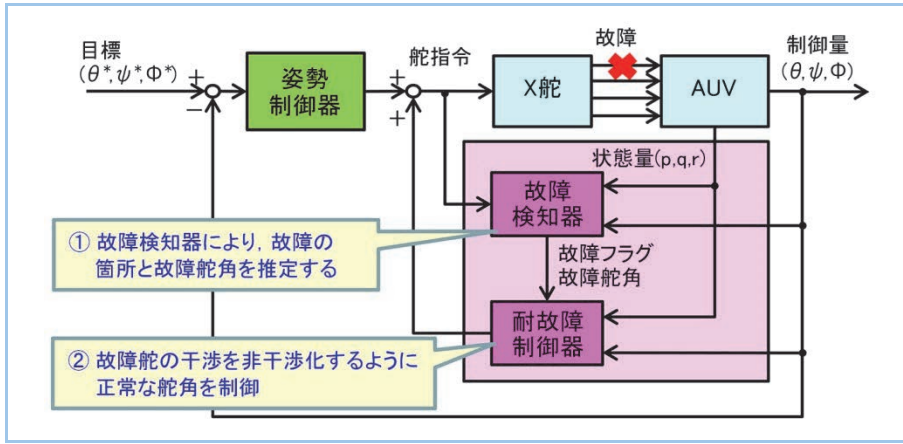


図3 耐故障制御器のブロック線図

3.1 故障検知器の開発

図4に故障検知器のブロック線図を示す。故障検知器は、無人ビークルの入力と出力からアクチュエータの故障箇所と故障の大きさを推定する。ここで、無人ビークルの姿勢運動をセンシングした結果と機体の運動モデルを用いて、状態を推定できるオブザーバを用いることが一般的であるが、図1に示すようなアクチュエータが冗長に配置されている場合、推定するアクチュエータの数4は状態量の数3よりも多いため、単一のオブザーバでは、操作量を精度よく推定することができない。

そこで、ある1つのアクチュエータが正常であると仮定し、他のアクチュエータ操作量を推定するオブザーバを各アクチュエータに対して構築する。これにより、オブザーバが推定対象とする操作量を縮退させることができ、推定するアクチュエータ操作量の数は状態量の数と同一になる。正常と仮定したアクチュエータは、制御器からの入力量をアクチュエータ操作量として扱う。

例として舵#1 を正常と仮定した場合の推定対象となる系を式(2)に示す。ここで、 δr_i^{cmd} は舵#iの制御入力を表す。

$$\frac{d}{dt} \begin{bmatrix} \Delta p \\ \Delta q \\ \Delta r \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} \Delta p \\ \Delta q \\ \Delta r \end{bmatrix} + \begin{bmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \\ e_{31} & e_{32} \end{bmatrix} \begin{bmatrix} \Delta \phi \\ \Delta \theta \end{bmatrix} + \begin{bmatrix} b_{12} & b_{13} & b_{14} \\ b_{22} & b_{23} & b_{24} \\ b_{32} & b_{33} & b_{34} \end{bmatrix} \begin{bmatrix} \Delta \delta r_2 \\ \Delta \delta r_3 \\ \Delta \delta r_4 \end{bmatrix} + \begin{bmatrix} b_{11} \\ b_{21} \\ b_{31} \end{bmatrix} \Delta \delta r_1^{cmd} \quad (2)$$

式(2)において、舵#2, #3, #4 に係る右辺3項目の係数のランクは3となるため、疑似逆行列を用いて、舵の操作量を推定できる。

図4に示す通り、構成したオブザーバで推定した操作量のうち、同じアクチュエータに対して推定した複数の操作量を比較することで、故障したアクチュエータを判定する。例えば、舵#1 についての推定値は、舵#2 を正常と仮定し推定した $\widehat{\delta r}_1^2$ 、舵#3 を正常と仮定し推定した $\widehat{\delta r}_1^3$ 、舵#4 を正常と仮定し推定した $\widehat{\delta r}_1^4$ がある。ここで、舵#2, #3, #4 のいずれかに故障が発生した場合、制御入力と舵操作量に差が生じるため、故障舵を正常と仮定して推定した操作量には、他の推定値との偏差が発生すると考えられる。

そこで、各アクチュエータに対する故障判定ロジックは3つのオブザーバから出力される推定操作量の平均値から、最も偏差の大きいアクチュエータを故障候補とする。例として、図5に舵#1 に関する故障判定ロジックのフローチャートを示す。

最後に、3つの故障判定ロジックから出力される故障候補フラグの全会一致を持って、アクチュエータ故障を判定する。また、故障推定値は、オブザーバの推定誤差を考慮して推定操作量の平均値とした。

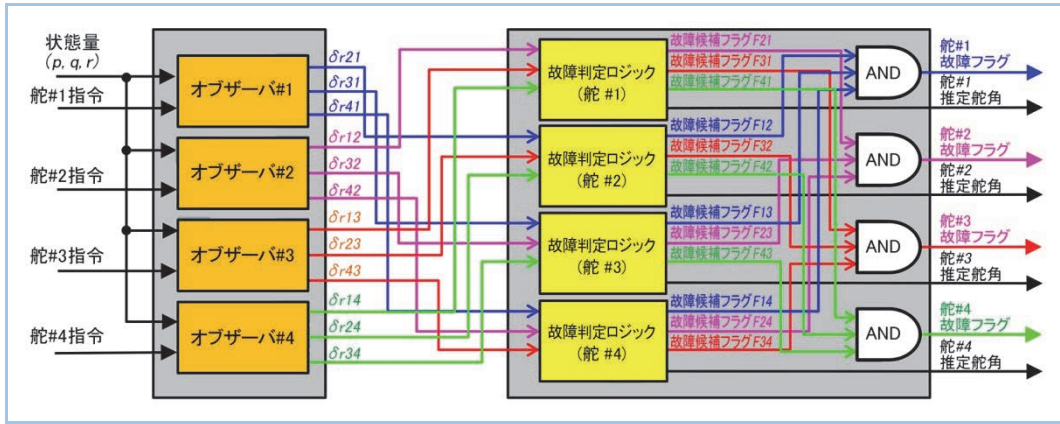


図4 故障検知器のブロック線図

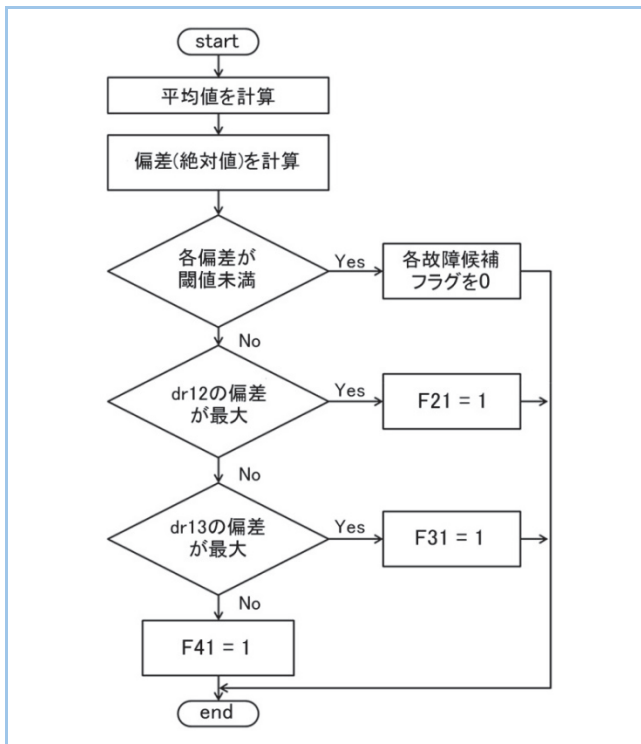


図5 故障判定ロジックのフローチャート(例:舵#1故障)

3.2 耐故障制御器の開発

非干渉制御器は、故障検知器から出力される故障アクチュエータの推定結果に基づいて、故障したアクチュエータが及ぼす機体への影響を打ち消すように、残存するアクチュエータの指令値を補正する非干渉制御則としている。

例えば、1番舵が故障した場合のAUVの状態方程式を式(3)に示す。故障舵は制御不能となり各運動軸に対して外乱力として作用し、制御可能なアクチュエータは3つとなり、3項目の係数が縮退される。

$$\frac{d}{dt} \begin{bmatrix} \Delta p \\ \Delta q \\ \Delta r \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} \Delta p \\ \Delta q \\ \Delta r \end{bmatrix} + \begin{bmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \\ e_{31} & e_{32} \end{bmatrix} \begin{bmatrix} \Delta \phi \\ \Delta \theta \end{bmatrix} + \begin{bmatrix} b_{12} & b_{13} & b_{14} \\ b_{22} & b_{23} & b_{24} \\ b_{32} & b_{33} & b_{34} \end{bmatrix} \begin{bmatrix} \Delta \delta r_2 \\ \Delta \delta r_3 \\ \Delta \delta r_4 \end{bmatrix} + \begin{bmatrix} b_{11} \\ b_{21} \\ b_{31} \end{bmatrix} \Delta \delta r_1^{failed} \tag{3}$$

故障アクチュエータの非干渉化のため、式(3)の運動軸(ロール角, ピッチ角, ヨー角)に関して、加角速度を0となるように変形し正常なアクチュエータについて解くことで、舵角補正量を求めることができる。

4. シミュレーション検証

4.1 シミュレーション条件

AUV を例題として、開発した耐故障制御の効果を検証した。シミュレーション条件として、AUV が直進航行中に、舵に故障が発生する場合を考える。なお、検証に用いるアクチュエータの故障は、直進航行中の AUV が備えるX舵のうち1枚が、外力により固着した条件とした。

4.2 シミュレーション結果

図6にシミュレーション結果の3次元航跡を示す。シミュレーション結果から耐故障制御を適用しない場合(赤線)には、アクチュエータ故障の影響により一度大きく浮上し、目標軌道から大きく逸脱した航跡となることが分かる。耐故障制御を適用した場合(青線)には、故障の影響を抑え、故障発生後も目標軌道に追従できている。

図7にシミュレーション結果の AUV 位置、および AUV 姿勢角を示す。耐故障制御がない場合には故障の影響により、y軸方向、z軸方向ともに大きく目標軌道から逸脱している。一方で、耐故障制御を適用することで、目標偏差を低減できることが確認できる。

図8にシミュレーション結果の舵操作量を示す。故障検知器により舵#1 の故障を即座に検知し、耐故障制御器により正常な舵で故障の影響を打ち消していることが確認できる。

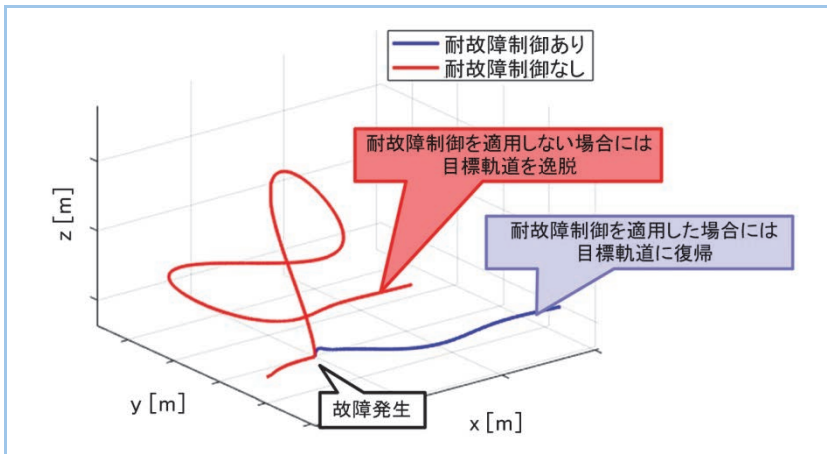


図6 シミュレーション結果(3D 航跡図)

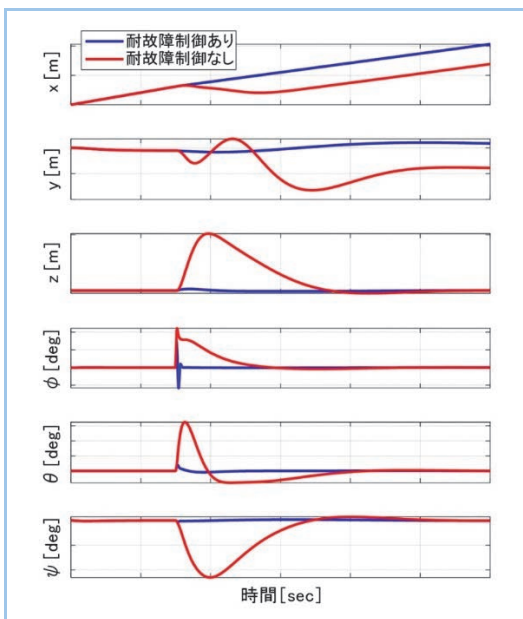


図7 シミュレーション結果(位置・姿勢角)

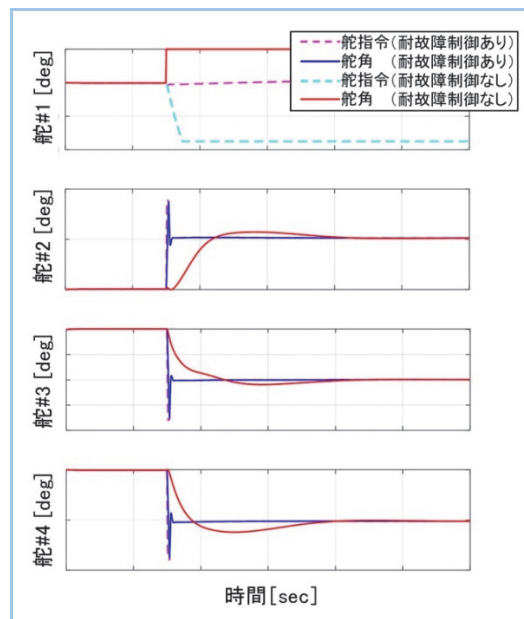


図8 シミュレーション結果(舵角)

5. まとめ

アクチュエータに冗長性を有している無人ビークルを対象として、各アクチュエータに対してオブザーバを構築することで構成される故障検知器と、故障アクチュエータの影響を正常なアクチュエータで打ち消すことで故障の影響を低減する耐故障制御器を開発した。

AUV を例題とした舵の固着故障のシミュレーションの結果、耐故障制御を適用することで、故障発生時の制御偏差を低減できることを確認し、開発した耐故障制御がアクチュエータ故障に対して有効であることが確認できた。

本報の成果は、アクチュエータに冗長性を有しているビークル製品 (AGV, UAV, AUV) のレジリエンス性能の向上に貢献できるものと考えられ、当社製品への適用を推進していく。

参考文献

- (1) 中西洋喜, 無人システムに求められる“強さ”, 計測と制御, 60 巻, 4号, (2021)p.248-249
- (2) 中村昌彦ほか, X舵を持つ AUV“ゆめいるか”の舵故障時の運動制御性能, 日本船舶海洋工学会論文集, 29 巻, (2019)p.103-115
- (3) Didier, Alexandre, et al. “Robust Adaptive Model Predictive Control of Quadrotors.” arXiv preprint arXiv:2102.13544 (2021)
- (4) 金井 喜美雄:フェイルセーフと制御システム設計, 計測と制御, 29 巻, 2号(1990)p.173-181
- (5) Zhenyu Yang, et al, Adaptive Control Mixer Method for Nonlinear Control Reconfiguration:A Case Study, IFAC Proceedings Volumes, Volume 33, Pages 557-562, Issue 15 (2000)
- (6) Yang, Zhenyu, et al, Robust control mixer method for reconfigurable control design using model matching, IET Control Theory & Applications 1.1 (2007), p.349-357
- (7) 染谷 歩, 大森 浩充:確定外乱とアクチュエータ故障を考慮した適応追従制御系の一設計法, 電気学会研究会資料, 産業計測制御研究会, 2006 巻, 114 号 (2006), p.41-46
- (8) 小田一道ほか, アクチュエータの冗長性を利用した無人移動体の耐故障制御の開発, 第9回計測自動制御学会制御部門マルチシンポジウム(2022)