

宇宙航空研究開発機構研究開発報告

JAXA Research and Development Report

風洞基準圧力計測配管の圧力応答シミュレーションによる最適化

上野 真, 須谷 記和, 香西 政孝
我那覇 義人, 今 直樹

2006年3月

宇宙航空研究開発機構
Japan Aerospace Exploration Agency

宇宙航空研究開発機構研究開発報告
JAXA Research and Development Report

風洞基準圧力計測配管の圧力応答シミュレーションによる最適化
Optimization of Wind Tunnel Reference Pressure Measurement Tubes
by Dynamic Response Simulations

上野 真^{*1}、須谷 記和^{*1}、香西 政孝^{*1}、我那覇 義人^{*2}、今 直樹^{*2}

Makoto UENO^{*1}, Norikazu SUDANI^{*1}, Masataka KOHZAI^{*1}, Yoshihito GANAHA^{*2}, Naoki KON^{*2}

*1：総合技術研究本部 風洞技術開発センター
Wind Tunnel Technology Center
Institute of Aerospace Technology

*2：株式会社コスモテック
COSMOTEC Inc.

2006年3月
March 2006

宇宙航空研究開発機構
Japan Aerospace Exploration Agency

目 次

概要	1
Nomenclature	1
1 はじめに	3
2 圧力計測と配管	3
2.1 総圧と静圧の位相差	3
2.2 配管応答の評価	4
3 圧力応答の定式化	4
3.1 配管 - センサー系モデル	4
3.2 モデルの定式化	5
3.3 プログラムの作成	7
4 ステップ応答実験	8
4.1 実験系	8
4.2 実験データ	9
5 解析の例	13
5.1 配管の組み合わせ	13
5.2 配管の狭窄	14
6 P_0 、 P_∞ 計測用新規配管	14
6.1 チャートの作成	14
6.2 response time	15
6.3 settling time	15
6.4 新規配管の選択	15
7 おわりに	18
8 謝辞	18
A 作成したプログラムと使用方法	19
A.1 概要	19
A.2 使用方法	19
B 使用上の注意	31
C ソースコード	32
参考文献	50

風洞基準圧力計測配管の 圧力応答シミュレーションによる最適化

上野 真*, 須谷 記和*, 香西 政孝*, 我那覇 義人†, 今 直樹†

Optimization of Wind Tunnel Reference Pressure Measurement Tubes by Dynamic Response Simulations

Makoto UENO*, Norikazu SUDANI*, Masataka KOHZAI*,
Yoshihito GANAHA† and Naoki KON†

abstract

The JAXA 2m×2m transonic wind tunnel utilizes total pressure and static pressure measured in the settling chamber and the plenum chamber respectively to compute Mach number. Very long period fluctuations of these pressures occur and it had been thought that such changes did not exert a significant influence on Mach number measurement. However, the pressure measurement systems employed very long tubes and it transpired that the phase difference between two measured pressures made the Mach numbers fluctuate. Consequently, the pipes having an optimal length and the inner diameter were designed and installed. To carry out this optimization, a computer program was developed that is capable of computing the response characteristics of pressure measurement systems without flow. The source code of the program is appended to the end of the report.

概要

宇宙航空研究開発機構の2m×2m遷音速風洞は集合胴総圧とプリナム室静圧を利用して測定部のマッハ数を算出しているが、これらの圧力は定常運転時も微小で長周期の変動をしている。従来、このような変動はマッハ数計測に有意な影響を及ぼさないと考えられてきたが、非常に長い配管を使用して計測しているため、配管を伝わる変動の位相差によってマッハ数に変動が現れる可能性があることが判明した。そこで、新たに長さや太さを最適化した総圧・静圧計測用配管を敷設することとした。このために、太さや長さの違う配管を接続して圧力孔からセンサーまで圧力を導いた圧力計測システムの応答特性を流れ場の無い場合において簡便に計算できるソフトウェアを開発し、コンピュータ・シミュレーションによって最適化した配管を敷設することができた。巻末には作成したプログラムのソースコードを添付し読者の便を図ることとした。

Keywords : Dynamic Pressure Measurement, Response Characteristics, Wind Tunnel

Nomenclature

$$a_0 = \sqrt{\frac{\gamma p_s}{\rho_s}} \quad \text{配管内の平均音速}$$
$$C_p \quad \text{等圧比熱}$$

*総合技術研究本部 風洞技術開発センター (Wind Tunnel Technology Center, Institute of Aerospace Technology)

† (株) コスモテック (COSMOTEC Inc.)

C_v	等容比熱
$D = 2R$	配管内径
f	周波数
g	重力加速度
$i = \sqrt{-1}$	虚数単位
J_n	n 次の第一種ベッセル関数
k	ポリトロープ指数
$\bar{k} = \frac{\nu R}{a_0}$	無次元周波数
L	配管長
M	マッハ数
m	質量流量
N	接続された配管の本数
$n = \left[1 + \frac{\gamma-1}{\gamma} \frac{J_2(i^{3/2}(\sqrt{\text{Pr}})s)}{J_0(i^{3/2}(\sqrt{\text{Pr}})s)} \right]^{-1}$	配管内部のポリトロープ指数
P	測定対象圧力
P_0	風洞総圧
P_∞	風洞静圧
$\text{Pr} = \frac{\mu C_p}{\lambda}$	プラントル数
p	p_s で無次元化された微小圧力変動量
p_s	平均圧力
q_∞	動圧
R	配管半径
R_0	気体定数
r	配管内円柱座標系の半径方向座標
$s = R\sqrt{\frac{\rho_s \nu}{\mu}}$	剪断波数 (shear wave number)
T	T_s で無次元化された微小温度変動量
T_s	平均温度
t	時間
u	x 方向速度成分の微小変動量
V_v	圧力センサー内部の気室体積
$V_t = \pi R^2 L$	配管体積
v	r 方向速度成分の微小変動量
x	配管内円柱座標系の流れ方向座標
$\gamma = \frac{C_p}{C_v}$	比熱比
$\Gamma = \sqrt{\frac{J_0(i^{3/2}s)}{J_2(i^{3/2}s)} \frac{\gamma}{n}}$	
$\eta = \frac{r}{R}$	配管内半径方向の無次元座標系
λ	熱伝導率
μ	粘性係数
$\nu = 2\pi f$	角振動数
$\xi = \frac{\omega}{a_0} x$	配管内流れ方向の無次元座標系
ρ_s	平均密度
ρ	ρ_s で無次元化された密度の微小変動量

σ 圧力センサー内部の気室体積のダイアフラム変形による増加量 (V_v で無次元化)

$$\phi = \frac{\nu \Gamma}{a_0}$$

Subscripts

j j 番目の配管またはセンサー
 v センサー内部の気室の体積
 t 配管の体積

1 はじめに

近年、原油価格の高騰や競争の激化に伴うコスト削減要求に対応して風洞試験の出力に対する精度向上要求は厳しくなる傾向にある。圧力は天秤出力と並んで風洞試験の主要な計測値の一つであり、マッハ数の決定や、抵抗の測定精度に影響を与えるベース圧力の測定など、風洞出力の精度に対する大きな影響を持っている。

宇宙航空研究開発機構（以下「JAXA」）の 2m×2m 遷音速風洞（以下、「TWT」）は連続風洞であるため、定常運転中も総圧 P_0 （及び静圧 P_∞ ）が 80～160 秒前後の周期で微小な変動をしている。従来はこのような微小な変動の影響は無視できるとされてきたため、圧力計測点と実際にセンサが設置されている位置は 20m 以上の距離が有り、また、圧力配管の途中に分岐が設けられていたりする。

しかし、マッハ数出力のさらなる高精度化を目指す過程で、この微小変動の影響が無視できない可能性が出てきた。まず、配管が長いことにより、測定部（模型設置位置）の P_∞ に対して、センサの計測する P_∞ は常に遅れている。また、 P_0 と P_∞ の計測圧力配管の長さや分岐の状態が同一ではないため、 P_0 と P_∞ の計測圧力に位相のずれが発生している。圧力変動は微小ではあるが、従来の配管で計測した P_∞ を使用すると、算出マッハ数が ± 0.002 程度の誤差を示すことが判明した。

圧力の計測は通常、測定点に開けた圧力孔から圧力配管で測定点の圧力を圧力センサーまで導いて計測するため、 P_∞ 測定のみならず、TWT における圧力計測のほとんどすべてにおいて配管の影響は避けられない。実際、ベース圧力のような圧力は通常

$$C_p = \frac{P - P_\infty}{q_\infty}$$

で表されるように P_∞ との差分を主流の動圧で無次元化して使用するため、計測圧力 P と P_∞ が同期していなければ精度は落ちる。

以上のような状況から TWT では既存の配管を見直す必要があるとの判断がなされたが、圧力配管を見直すとしても現状の分析や許容範囲の設定などのために配管系がどのような圧力応答特性を持つのか知る必要がある。

本文書では配管見直しの過程で行われた配管応答の評価および評価用に作成されたプログラムと見直しの結果について述べる。

2 圧力計測と配管

2.1 総圧と静圧の位相差

TWT では、測定部上流の集合胴で P_0 を計測し、測定部を取り囲むプリナム室内部で P_∞ を計測している。従来、 P_0 、 P_∞ を計測する配管の太さや長さには特に注意が払われてこなかった。これは風洞の P_0 、

P_∞ の定常運転中の変動は微小であり、変動があったとしてもその周波数は小さいので、 P_0 と P_∞ から算出されるマッハ数に対する影響は無視できると考えられてきたからであると推測されている。

しかし、マッハ数の計測精度を精査する過程で、 P_0 と P_∞ の変動は位相がずれており、この位相ずれによって生じる算出マッハ数の誤差には改善の余地があることが分かった。この誤差量は peak to peak で 0.002 程度である。

マッハ数 0.9 の時の P_0 と P_∞ の変動量は最大でも 240 Pa (peak to peak) 程度、実際の位相のずれは数秒程度であるが、供試体によっては抵抗係数に換算して 0.0003 (3 ドラッグカウント) 以上の誤差をもたらす場合があると試算された。

2.2 配管応答の評価

風洞内の配管は設備上の制約から圧力応答を実測することが困難である。また、既存の配管は敷設資料が残っておらず、容易には計測不可能な位置で複雑に分岐されているため、既存の敷設経路の詳細を明らかにすることは断念した。その代わりに、新たに分岐の P_0 、 P_∞ 計測用圧力配管を新設することを決め、新設する配管は圧力応答をシミュレーションで評価して決定することとした。従って、実験的に新規配管の有効性を検証することはできないが、明確に長さや太さの情報が決定された配管を使用して遅れがほとんど無いと想定される P_0 、 P_∞ の計測を行えるようにすることを目指した。

圧力配管の応答に関しては風洞での非定常圧力計測に関する Bergh らの研究 [1, 2] がある。

Bergh らは配管末端から入力された微小振動の振幅と位相が配管の他端でどのように変化するかを配管内の Navier-Stokes 方程式を微小振動入力に対して解くことによって解析した [1, 2]。この手法の利点は、応答の解析解が得られており容易に周波数応答を得られることであり、文献 [1] 内で詳細な実験的検証が行われている。また、Chapin [3] によって追試が行われており、配管をらせん状にしたりして極端な曲率を与えなければ有効であるとの結論が出されている。

ただし、Tijdeman は風洞試験模型翼表面の圧力孔のように圧力孔が流れにさらされているときは、圧力応答が無風のときと異なることを指摘している [4, 2]。本文書で扱うのは流れのよどんでいる P_0 と流れの緩やかなプリナム室内で計測される P_∞ なので、Bergh らの研究結果を使用することに問題は無いと判断した。

以上より、Bergh らの研究結果を利用して配管応答の評価を行い、その結果に基づいて新規に圧力計測用配管を敷設することとした。以下の節でその概要を述べる。

3 圧力応答の定式化

3.1 配管-センサー系モデル

まず、1 本の配管と 1 つのセンサーからなる系を図 1 のように表す。ここではセンサーは一つの気室として考えられており、図中右側のセンサーに内径 $2R$ で長さ L の管が接続されている。配管入り口における平均圧力が p_s であるとき、左側から $p_s p_0 e^{i\omega t}$ なる微小変動入力を加えた場合のセンサー内部の圧力、すなわちセンサーの計測圧力を $p_s(1 + p_1 e^{i\omega t})$ で表す。ただし、圧力センサーは端部のダイアフラムの変形量を計測して圧力に変換していることも多いため、入力圧力の変動によってセンサー内部の容積も変化すると、気室容積は $V_{v1}(1 + \sigma_1 p_1 e^{i\omega t})$ で表す。ここで、 σ はダイアフラムの変形によるセンサー内部容積の無次元増加量であり、通常微小であるが、無視できない場合は実験などで決定する必要がある。

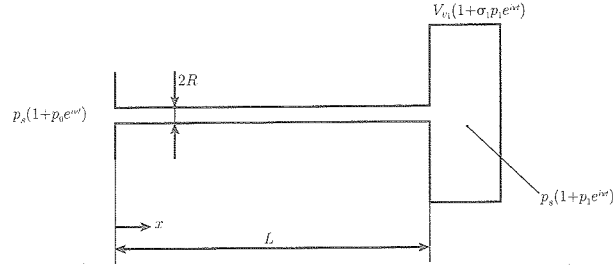


図 1: 1本の配管とセンサーからなる圧力計測系。

複数の太さや長さの配管が接続された配管系は図 1 のような系が連続したものであると考えると図 2 のようになる。モデル化の都合上配管接続部に気室（センサー）が存在することになっているが、通常の配管系ではこれらの気室の容積は 0 と考えれば良い。

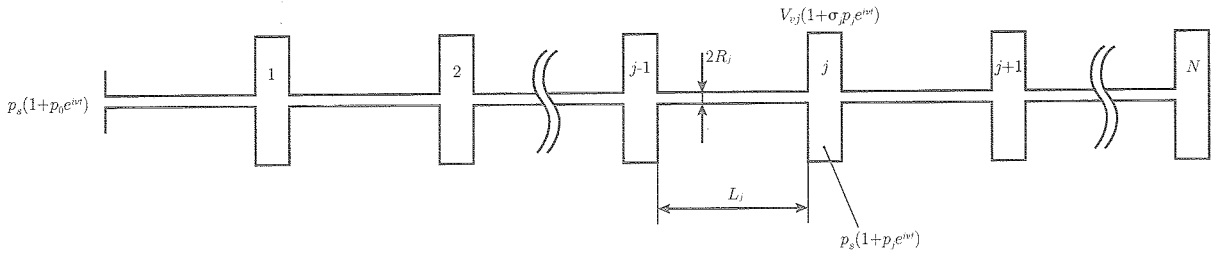


図 2: 複数の接続配管からなる圧力計測系。

3.2 モデルの定式化

配管内部の座標系として図 3 のように円筒座標系を設定する。この配管内部の気体の挙動を記述する方程式は以下になる。

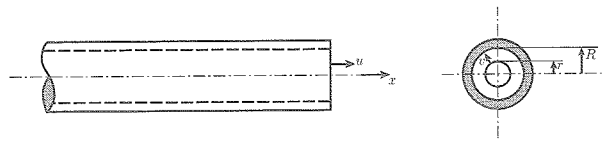


図 3: 配管内部の座標系。

● Navier-Stokes 方程式

$$\bar{\rho} \left(\frac{\partial \bar{u}}{\partial t} + \bar{v} \frac{\partial \bar{u}}{\partial r} + \bar{u} \frac{\partial \bar{u}}{\partial x} \right) = -\frac{\partial \bar{p}}{\partial x} + \mu \left\{ \left(\frac{\partial^2 \bar{u}}{\partial x^2} + \frac{\partial^2 \bar{u}}{\partial r^2} + \frac{1}{r} \frac{\partial \bar{u}}{\partial r} \right) + \frac{1}{3} \frac{\partial}{\partial x} \left(\frac{\partial \bar{u}}{\partial x} + \frac{\partial \bar{v}}{\partial r} + \frac{\bar{v}}{r} \right) \right\} \quad (3.1a)$$

$$\bar{\rho} \left(\frac{\partial \bar{v}}{\partial t} + \bar{v} \frac{\partial \bar{v}}{\partial r} + \bar{u} \frac{\partial \bar{v}}{\partial x} \right) = -\frac{\partial \bar{p}}{\partial r} + \mu \left\{ \left(\frac{\partial^2 \bar{v}}{\partial r^2} + \frac{1}{r} \frac{\partial \bar{v}}{\partial r} - \frac{\bar{v}}{r^2} + \frac{\partial^2 \bar{v}}{\partial x^2} \right) + \frac{1}{3} \frac{\partial}{\partial r} \left(\frac{\partial \bar{u}}{\partial x} + \frac{\partial \bar{v}}{\partial r} + \frac{\bar{v}}{r} \right) \right\} \quad (3.1b)$$

● 連続の式

$$\frac{\partial \bar{\rho}}{\partial t} + \bar{u} \frac{\partial \bar{\rho}}{\partial x} + \bar{v} \frac{\partial \bar{\rho}}{\partial r} + \bar{\rho} \left[\frac{\partial \bar{u}}{\partial x} + \frac{\partial \bar{v}}{\partial r} + \frac{\bar{v}}{r} \right] \quad (3.2)$$

● 理想気体の状態方程式

$$\bar{p} = \bar{\rho} R_0 \bar{T} \quad (3.3)$$

● エネルギー式

$$\rho C_p \left[\frac{\partial \bar{T}}{\partial t} + \bar{u} \frac{\partial \bar{T}}{\partial x} + \bar{v} \frac{\partial \bar{T}}{\partial r} \right] = \lambda \left[\frac{\partial^2 \bar{T}}{\partial r^2} + \frac{1}{r} \frac{\partial^2 \bar{T}}{\partial x^2} \right] + \frac{\partial \bar{p}}{\partial t} + \bar{u} \frac{\partial \bar{p}}{\partial x} + \bar{v} \frac{\partial \bar{p}}{\partial r} + \mu \Phi \quad (3.4a)$$

$$\Phi = 2 \left[\left(\frac{\partial \bar{u}}{\partial x} \right)^2 + \left(\frac{\partial \bar{v}}{\partial r} \right)^2 + \left(\frac{\bar{v}}{r} \right)^2 \right] + \left[\frac{\partial \bar{v}}{\partial x} + \frac{\partial \bar{u}}{\partial r} \right]^2 - \frac{2}{3} \left[\frac{\partial \bar{u}}{\partial x} + \frac{\partial \bar{v}}{\partial r} + \frac{\bar{v}}{r} \right]^2 \quad (3.4b)$$

入力として

$$\bar{u} = a_0 u(x, r) e^{i\nu t} \quad (3.5a)$$

$$\bar{v} = a_0 v(x, r) e^{i\nu t} \quad (3.5b)$$

$$\bar{p} = p_s \{1 + p(x, r) e^{i\nu t}\} = \{\rho_s a_0^2 / \gamma\} \{1 + p(x, r) e^{i\nu t}\} \quad (3.5c)$$

$$\bar{\rho} = \rho_s \{1 + \rho(x, r) e^{i\nu t}\} \quad (3.5d)$$

$$\bar{T} = T_s \{1 + T(x, r) e^{i\nu t}\} \quad (3.5e)$$

ここで、振動成分 u 、 v 、 p 、 ρ 、 T は十分小さいとし、以下の無次元座標系を導入すれば

$$\xi = \frac{\omega}{a_0} x, \quad \eta = \frac{r}{R} \quad (3.6)$$

上記の方程式 (3.1)～(3.4) は次のように簡略化される。

$$iu = -\frac{1}{\gamma} \frac{\partial p}{\partial \xi} + \frac{1}{s^2} \left[\left(\bar{k}^2 \frac{\partial^2 u}{\partial \xi^2} + \frac{\partial^2 u}{\partial \eta^2} + \frac{1}{\eta} \frac{\partial u}{\partial \eta} \right) + \frac{1}{3} \bar{k} \frac{\partial}{\partial \xi} \left(\bar{k} \frac{\partial u}{\partial \xi} + \frac{\partial v}{\partial \eta} + \frac{v}{\eta} \right) \right] \quad (3.7a)$$

$$i\bar{k}v = -\frac{1}{\gamma} \frac{\partial p}{\partial \eta} + \frac{\bar{k}}{s^2} \left[\left(\frac{\partial^2 v}{\partial \eta^2} + \frac{1}{\eta} \frac{\partial v}{\partial \eta} - \frac{v}{\eta^2} + \bar{k}^2 + \frac{\partial^2 v}{\partial \xi^2} \right) + \frac{1}{3} \frac{\partial}{\partial \eta} \left(\bar{k} \frac{\partial u}{\partial \xi} + \frac{\partial v}{\partial \eta} + \frac{v}{\eta} \right) \right] \quad (3.7b)$$

$$i\bar{k}\rho = -\left(\bar{k} \frac{\partial u}{\partial \xi} + \frac{\partial v}{\partial \eta} + \frac{v}{\eta} \right) \quad (3.7c)$$

$$p = \rho + T \quad (3.7d)$$

$$iT = i \frac{\gamma - 1}{\gamma} p + \frac{1}{\text{Pr} s^2} \left(\frac{\partial^2 T}{\partial \eta^2} + \frac{1}{\eta} \frac{\partial T}{\partial \eta} + \bar{k}^2 \frac{\partial^2 T}{\partial \xi^2} \right) \quad (3.7e)$$

これらの方程式を以下の条件の下で解くことによって配管内の圧力応答を求めることができる [2]。

- 管内の気体は均質であり、変動の波長と配管半径は平均自由行程に対して十分に大きい。この仮定は常温常圧の空気については $f > 10^8$ または $R < 10^{-5}$ で崩れる。
- 配管内に定常的な流れは無い。
- 配管は十分に長く、末端の効果は無視できる。

また、センサー内部の気室容積は十分に小さく、内部に圧力分布は生じないと考えることによって気室容積の影響も解に含むことができる。

方程式の解法の詳細については文献 [1, 2] に詳しいため、ここでは繰り返さない。

図 2 の配管 j と $j+1$ の間にある気室 j での圧力変動は

$$\frac{p_j}{p_{j-1}} = \left[\cosh(\phi_j L_j) + \frac{V_{vj}}{V_{tj}} \left(\sigma_j + \frac{1}{k} \right) \phi_j L_j n_j \sinh(\phi_j L_j) + \frac{V_{tj+1}}{V_{tj}} \frac{a_{0j}^2}{a_{0j+1}^2} \frac{\phi_j}{\phi_{j+1}} \frac{L_j}{L_{j+1}} \frac{n_j}{n_{j+1}} \frac{\sinh(\phi_j L_j)}{\sinh(\phi_{j+1} L_{j+1})} \left\{ \cosh(\phi_{j+1} L_{j+1}) - \frac{p_{j+1}}{p_j} \right\} \right]^{-1} \quad (3.8)$$

と表される。ここで、 ϕ は以下のように定義される

$$\phi_j = \frac{\nu \Gamma_j}{a_0}, \quad \Gamma_j = \sqrt{\frac{J_0(i^{3/2}s)}{J_2(i^{3/2}s)} \frac{\gamma}{n_j}}, \quad n_j = \left[1 + \frac{\gamma - 1}{\gamma} \frac{J_2(i^{3/2}s_j \sqrt{\text{Pr}})}{J_0(i^{3/2}s_j \sqrt{\text{Pr}})} \right]^{-1} \quad (3.9)$$

ただし、末端の配管 $j = N$ では、

$$\frac{p_N}{p_{N-1}} = \left[\cosh(\phi_N L_N) + \frac{V_{vN}}{V_{tN}} \left(\sigma_N + \frac{1}{k} \right) \phi_N L_N n_N \sinh(\phi_N L_N) \right]^{-1} \quad (3.10)$$

3.3 プログラムの作成

上述の式を再帰的に用いることによって複数の配管が接続された配管系の圧力応答を推算することができる。そこで、作業の効率化と今後の利便性を考慮して、汎用の配管圧力応答推算プログラムを作成し、評価に活用することとした。

作成したプログラムは

- 配管の内径
- 配管の長さ
- センサー内部の気室の容積
- センサー内部の気室の容積がダイアフラムの変形によって変化する割合

を入力として、長さや内径の違う配管を接続して構成された圧力計測系の

- ボード線図
- ステップ応答
- 任意入力に対する応答

を算出することができる。

このプログラムを使用すれば圧力計測系の設計時点で圧力配管の適・不適が予測できることから、圧力計測システムの最適化に寄与することが期待できる。

作成したプログラムの仕様・使用方法などの詳細は Appendix A に示す。また、読者の利便を図るためにソースコードを Appendix C に掲載した。プログラムの実装に当たって、開発言語には開発効率と広く普及していることを考慮して MATLAB^{®1}を使用した。

また、式 (3.8) では音速 a_{0j} が各配管ごとに設定できることになっているが、これは各配管での静温度 T_s を設定できるということになる。プログラムを作成するに当たって、配管システム全体で T_s は一定であるとした。

¹MATLAB[®] は MathWorks, Inc. の登録商標である。その他、本文中に記載されている会社名、製品名は、それぞれ各社の商標または登録商標である。

4 ステップ応答実験

本文書で作成したプログラムの基になった文献 [1] では、周期圧力を発生する装置を用いて周波数ごとの応答特性を調べる実験を行って、モデルの検証を行っている。

本文書では、プログラム使用の例題として Bergh らの解析結果 [1] を出力することによってプログラムのボード線図出力の妥当性を検証した (Appendix A)。そこで、本節ではステップ応答について検証を試みた実験結果を示す。

4.1 実験系

実験系としては図 4 のような系を組んだ。塩化ビニール・チューブ (以下「塩ビチューブ」) の両端に圧力センサー (Statham PM131 5psi) を接続し、一方の端は圧力コントローラ (Druck DPI510) を介した真空ポンプにも接続する。真空ポンプと圧力コントローラを用いて大気圧より 30kPa 低い圧力にチューブ内の圧力を調圧した後に塩ビチューブの接続を外すことによって、ステップ入力を与える。真空ポンプ内の容積がセンサ 1 の出力に悪影響を及ぼすことを避けるため、接続を外す前に真空ポンプ側の配管を封止した。差圧の 30kPa は圧力センサのレンジから設定した。

圧力センサ 1 の出力をトリガとし、圧力センサ 2 の出力を応答としてデータ取得装置 (NEC 三栄 オムニユース RA2300) で計測した。

使用した塩ビチューブは内径 1mm、2mm、6mm の三種類で、それぞれ 2m、5m、10m の長さのものを用意した。応答予測では圧力センサー内の気室の容積は 0 と仮定している。周囲圧力は大気圧 (101325Pa)、周囲温度は実験時の室内気温 26℃とした。また、塩ビチューブと圧力センサを接続する継手や接続用の短い塩ビチューブは長さや太さを計測して、接続された配管として考慮した形で応答予測を行った。

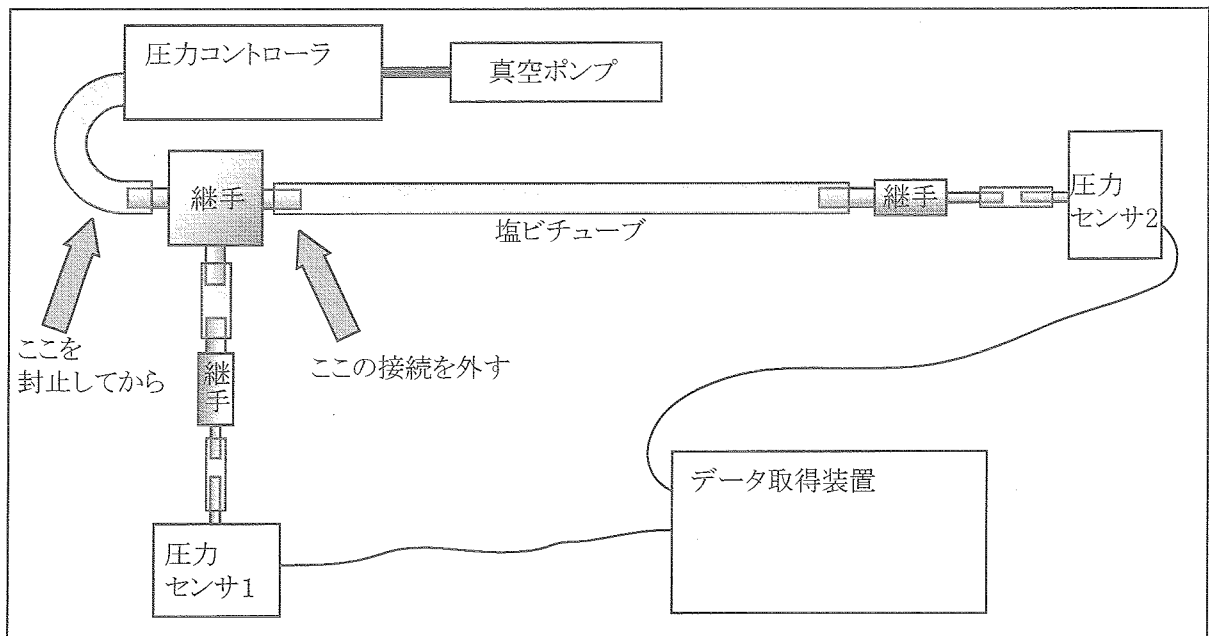


図 4: 実験系の模式図。

4.2 実験データ

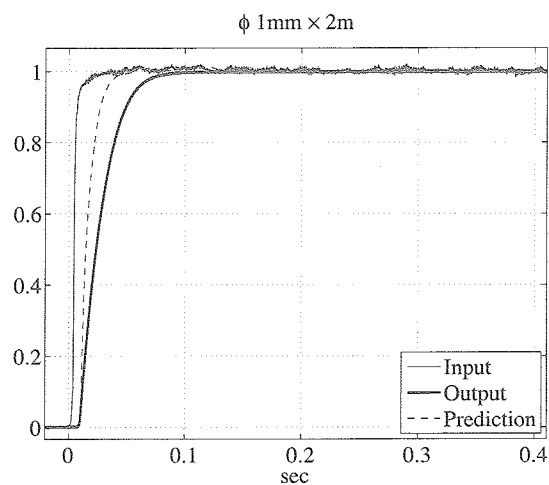
実験データを図 5～7 に示す。図中で実線が圧力センサ 1 の出力、太線が圧力センサ 2 の出力、点線が圧力センサ 2 の出力を予測したプログラムの出力である。また、図の縦軸はステップ入力前の圧力を 0 とし、ステップ応答後の終端圧力の差で正規化した出力である。

出力の傾向は以下の通りである。

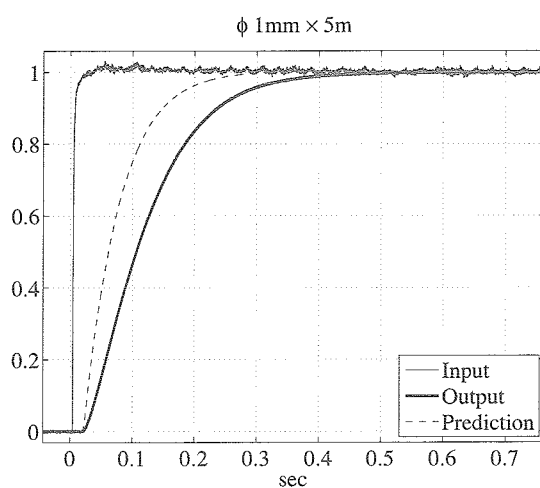
内径 1mm の配管	プログラム出力の方が実験よりも若干早い応答を示したが、応答速度のオーダーから言ってプログラム出力と実験の一致は良い。
内径 2mm の配管	予測と実験の一致の度合いは 1mm のケースと同様である。
内径 6mm の配管	短い配管ではプログラム出力と実験データが良く一致する。しかし、配管長が長くなるにつれて実験データの応答が立ち上がり途中で急激にプログラム出力と合わなくなる。

内径 6mm の配管ではステップ入力によって生じた衝撃波が配管の末端で反射して生じる出力の振動が観測されている。このような振動が無い、すなわち粘性によってステップ入力が入力がすぐに減衰してしまうような細くて長い配管、および、逆に強い振動出力が観測される太く短い配管（図 7(a)）ではプログラム出力と実験で観測された応答の一致は良い。

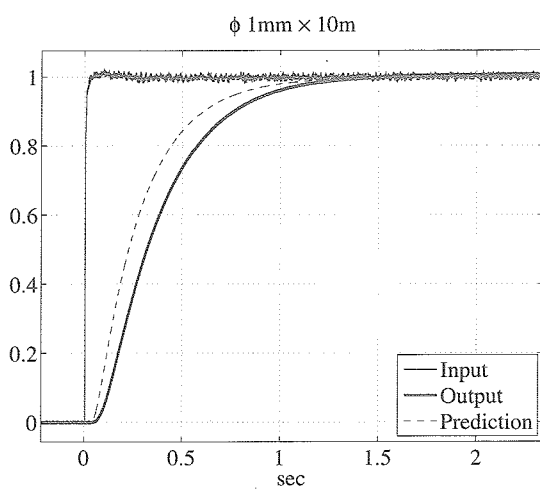
しかし、出力の振動がすぐに減衰してしまうような配管長と配管内径の組み合わせ（図 6(a), 7(b), 7(c)）では、プログラム出力と実験はあまり一致しなかった。これらの相違の原因としては、塩ビチューブを用いたことによる内径の微妙な変化や、曲率の影響が考えられる。特に、衝撃波の減衰は非線形な現象であるため、微妙な影響を受けやすい。管を巻くと減衰が強まることは Chapin [3] によっても指摘されており、このような配管では応答の予測特性は悪化する可能性がある。



(a) 長さ 2m

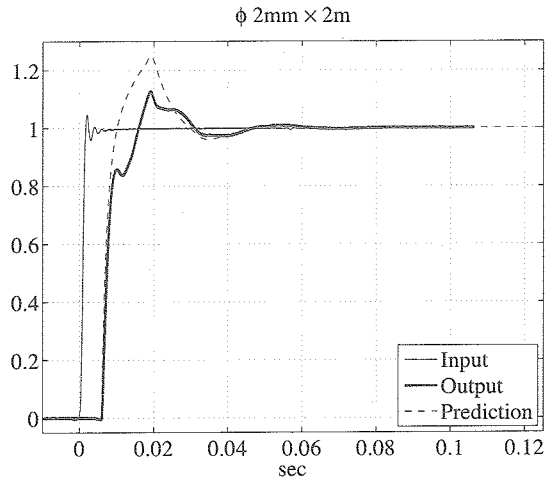


(b) 長さ 5m

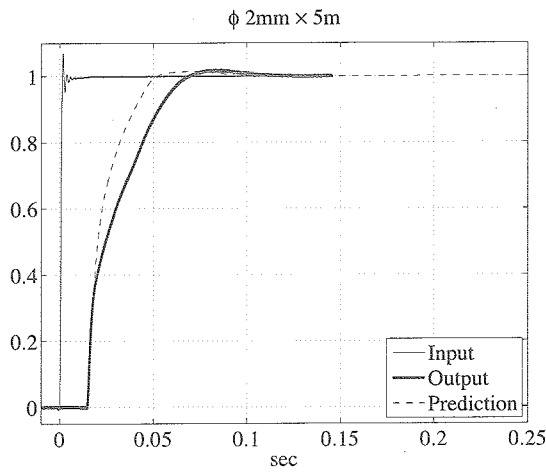


(c) 長さ 10m

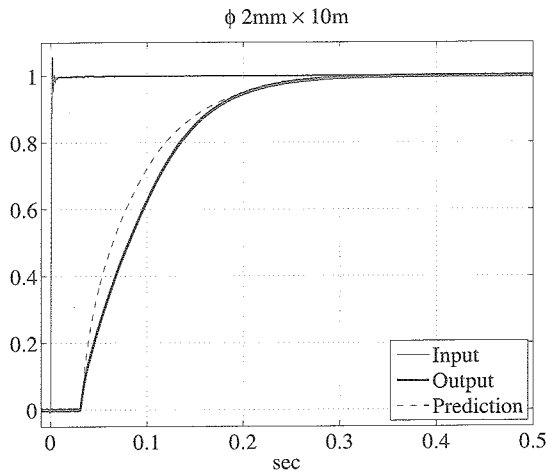
図 5: 配管内径が 1mm の場合.



(a) 長さ 2m

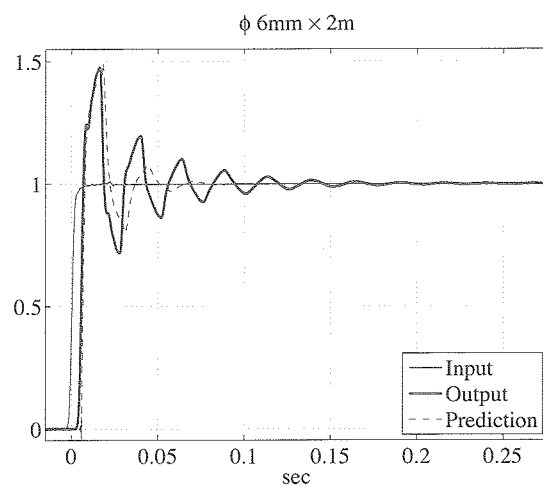


(b) 長さ 5m

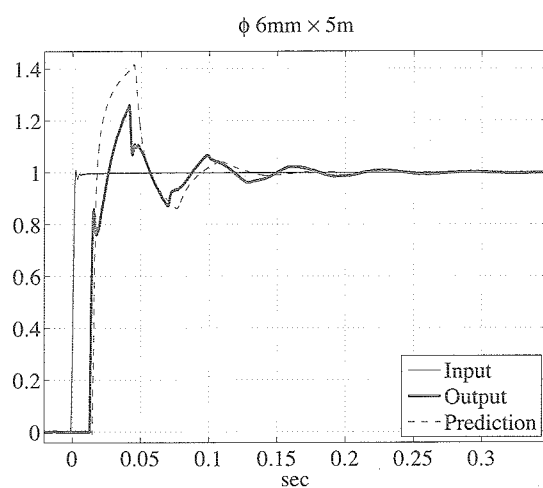


(c) 長さ 10m

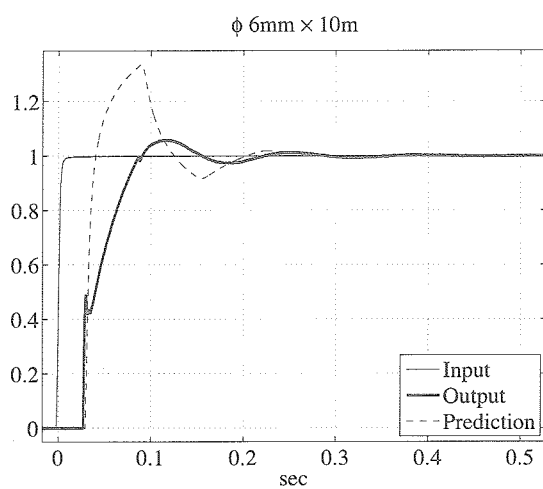
図 6: 配管内径が 2mm の場合.



(a) 長さ 2m



(b) 長さ 5m



(c) 長さ 10m

図 7: 配管内径が 6mm の場合.

5 解析の例

プログラムの活用事例を例を用いて示す。

5.1 配管の組み合わせ

風洞試験模型では内径 0.5mm の圧力孔から内径 1.0mm のビニールチューブに接続してセンサーまで接続するというようなことが多い。すなわち、細い配管から太い配管へと接続するわけだが、このような接続は圧力応答に対しては極めて不利である。ここではこのような場合について試算してみる。

図 8(a) のような 4 つのケースについて考える。いずれの配管も全長は 10m であるが、内径 0.5mm、1mm の二種類の配管を使用してちょうど全長の半分の 5m の位置で太さを変えた場合を想定した。

Case A 内径 0.5mm のまま

Case B 内径 0.5mm から 1.0mm の太い配管に接続した。

Case C 内径 1.0mm から 0.5mm の細い配管に接続した。

Case D 内径 1.0mm のまま

管の左端で入力したステップ入力が右端でどのように観測されるかを推算した結果が、図 8(b) である。

応答が最も早いのは当然予測されるように **Case D** であるが、**Case C** は応答速度がほとんど **Case D** と変わらない。また、**Case A** よりも **Case B** の方がはるかに遅い応答を示す。

すなわち、細い配管から太い配管に接続すると応答遅れが顕著になる (**Case B**)。これは太い配管から細い配管へと接続したとき (**Case C**) の応答が太い配管のままのときとほとんど変わらないのとは対照的である。

Case B では細い配管で入力が減衰した後に、太い配管内の容積の大きな気体を駆動しなければならないために遅れが生じるものと考えられる。

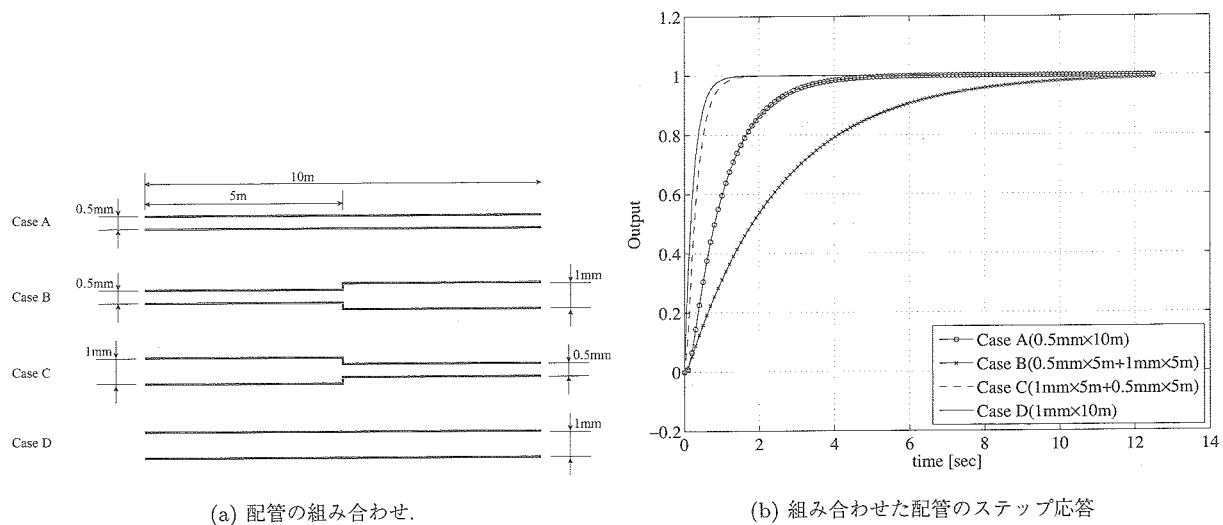


図 8: 配管の組み合わせの計算例.

5.2 配管の狭窄

ビニールチューブの長さが足りないとき、通常はニューマチックコネクタを使用して長さを足すことが多い。しかし、ニューマチックコネクタのような狭窄部を配管の途中に設けると深刻な応答遅れをもたらすことがある。本節では、配管途中に狭窄部を設けたときの応答を試算してみる。

図 9(a) のような 4 つのケースについて考える。いずれの配管も全長は 10m であるが、内径 0.5mm、1mm の二種類の配管を使用して Case F~Case H はそれぞれ長さ 1m の狭窄部を設けた。

図 9(b) にステップ応答の推算結果を示すが、狭窄部が配管の右端（出力側）に近いほど、遅れが出にくいことが分かる。逆に狭窄部が配管の左端（入力側）に近いと遅れは顕著になる。

これらのことから分かることは、圧力計測においてチューブの継ぎ足しはなるべく避けるべきであり、継ぎ足した後のチューブが長ければ長いほど応答は遅れるということである。

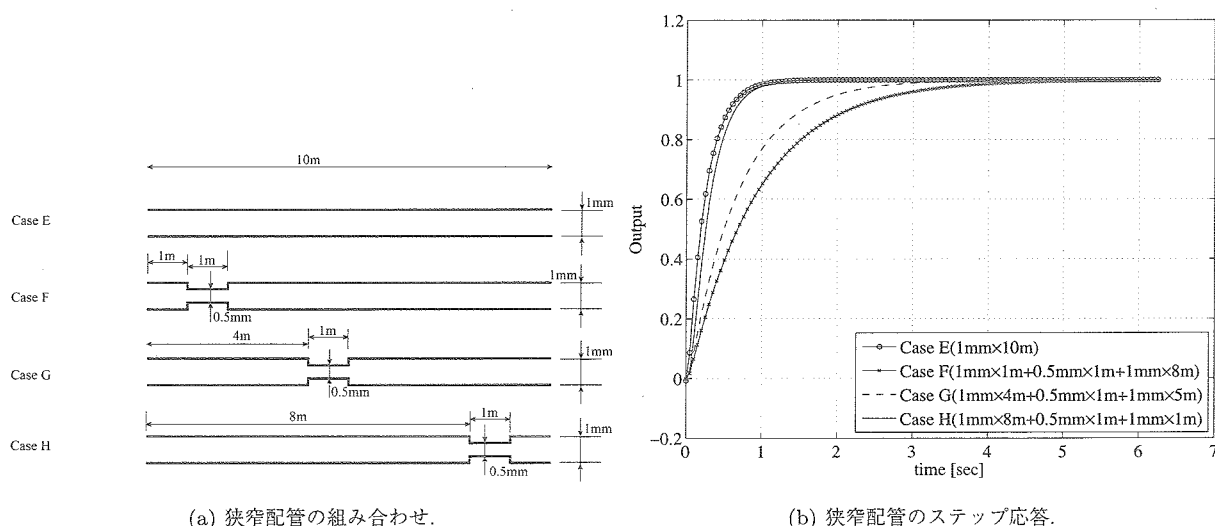


図 9: 配管の狭窄の計算例.

6 P_0 、 P_∞ 計測用新規配管

6.1 チャートの作成

4 節の結果から明かなように、細く長い配管では粘性の影響で遅れが顕著になり、太く短い配管では特定の周波数が増幅される。

そこで、配管の長さ及び太さと応答時間の関係を視覚的に表すことを試みた。長さや太さを変えた配管について最初に入力の 99% に達するまでの時間（response time とする）と、最終的に入力値の $\pm 1\%$ 以内に応答が入り、この時点以後は再びこの範囲を越えなくなるまでの時間（settling time とする）をステップ応答の結果を利用してチャートにした（図 10, 11）。

応答特性は入力の平均圧力と平均温度に依存する。TWT は P_0 を 50kPa から 150kPa、マッハ数 M を 0.0 および 0.1 から 1.4 までの範囲で設定することができる。 M が上がれば P_∞ は下がることになるので、 P_0 が低く、 M が高いときの P_∞ 計測値の平均値が最も低い圧力になる。また、その逆に P_0 が高く、 M が 0.0 であれば P_∞ （または P_0 ）計測値の平均値は最も高い圧力になる。

本節で扱うのは P_0 と P_∞ なので、これら二種類の圧力計測に対する入力圧力の下限值は $P_0 = 50\text{kPa}$ 、 $M=1.4$ のときの P_∞ 計測、上限値は $P_0 = 150\text{kPa}$ 、 $M=0.0$ のときの P_0 、 P_∞ 計測と考えられるため、こ

の2ケースについてチャートを作成することにした。

総温度は風洞内の標準的な総温度である 40°C に固定した。作動気体は乾燥空気である。

図中、白抜きのマーカーの点ではオーバーシュートが起こり、黒く塗りつぶしたマーカーの点ではオーバーシュートは起こっていない。評価した管の種類は表 2 に示す。

	入力圧力下限のケース	入力圧力上限のケース
総圧 P_0 [kPa]	50	150
静圧 P_{∞} [kPa]	15.7	150
マッハ数	1.4	0.0
総温度 [$^{\circ}\text{C}$]	40	40

表 1: 推算条件.

内径 [mm]	0.5, 1, 2, 3, 4, 6, 8, 10
管長 [m]	0.5, 1, 2, 5, 10, 20, 30

表 2: 評価した管の種類.

6.2 response time

response time は図 10(b), 11(b) からわかるように、管長を長くすると指数関数的に増大する。また、図 11(a) からわかるように内径を広げると response time は短くなるが、伝達速度は音速以上にはならないため、管長ごとに一定の値に収束する。静圧が低くなると応答に時間がかかり、管長によっては配管の内径を 10mm にしても十分な収束に達しない。(図 10(a))。

6.3 settling time

settling time も同様に管長が増大すると指数関数的に増大する (図 10(d), 11(d)) が、出力のオーバーシュートが起こらないときの増大率は著しい。また、オーバーシュートが起こる場合、配管の内径が増えるほど settling time は増大する (図 10(c), 11(c))。

6.4 新規配管の選択

TWT では総圧 P_0 及び静圧 P_{∞} の計測配管を最適化するために、新規の配管を敷設することを決定した。

基本的に遅れが発生すると位相ずれなどの問題が顕著になるため、配管を極力短くすることにした。検討の結果、配管長は最短で 5m まで短くすることができる。そこで、図 10, 11 より 5m の配管の応答を調べると、以下のようなことが分かる

- response time は内径 6mm 程度で収束する (図 10(a), 11(a))。ただし、オーバーシュートする。
- settling time については図 10 の条件では内径 4mm (図 10(c))、図 11 の条件では内径 2mm (図 11(c)) で最小となる。
- 内径 2mm の配管は条件によってオーバーシュートしたりしなかったりする。(図 10(c), 11(c))

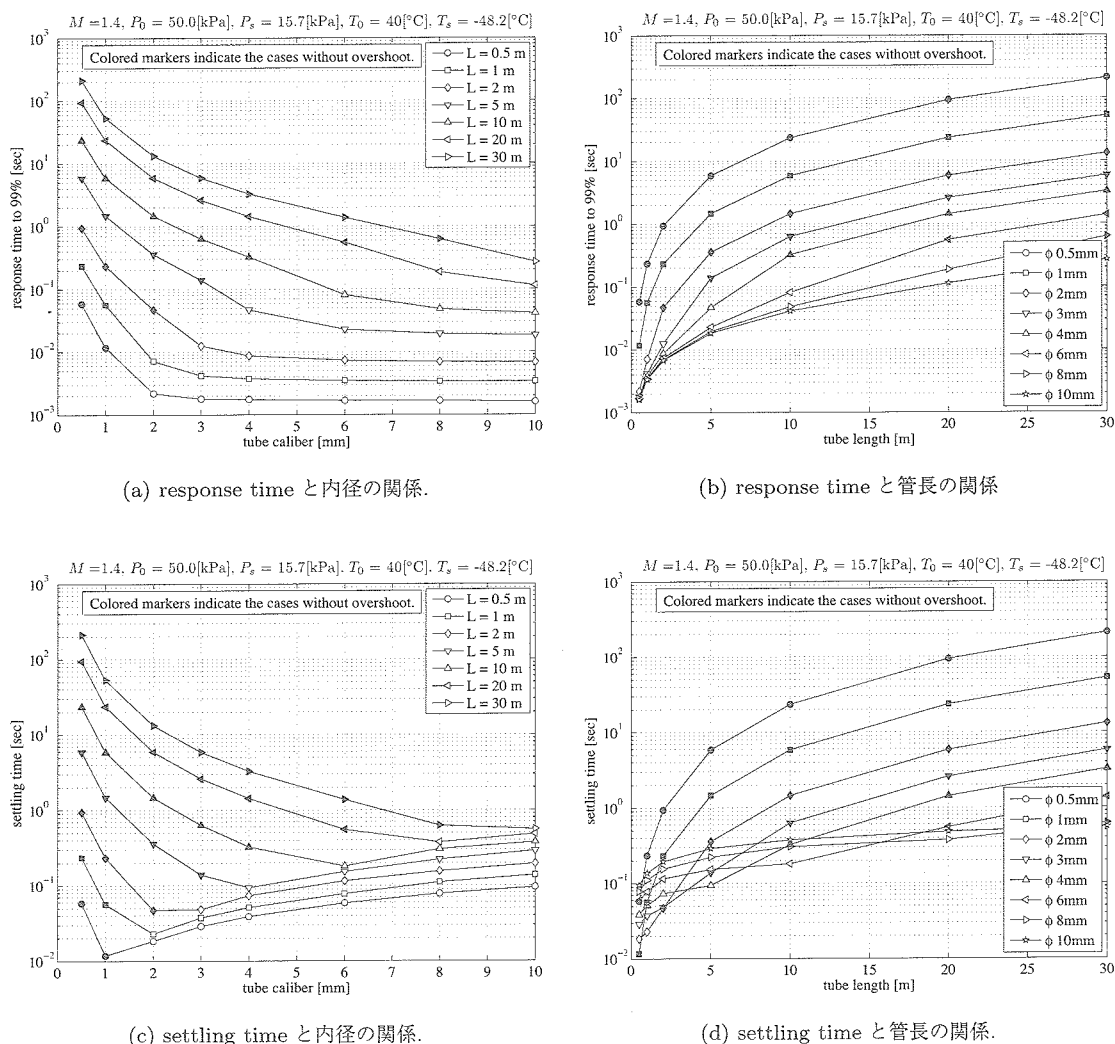
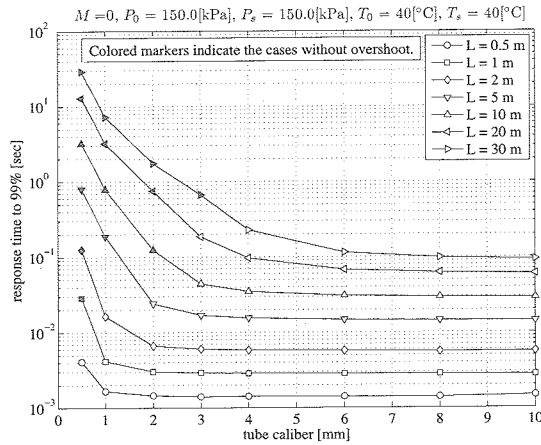


図 10: 内径と管長を変えたときの応答チャート ($M = 1.4$, $P_0 = 50$ kPa, $T_0 = 40$ °C).

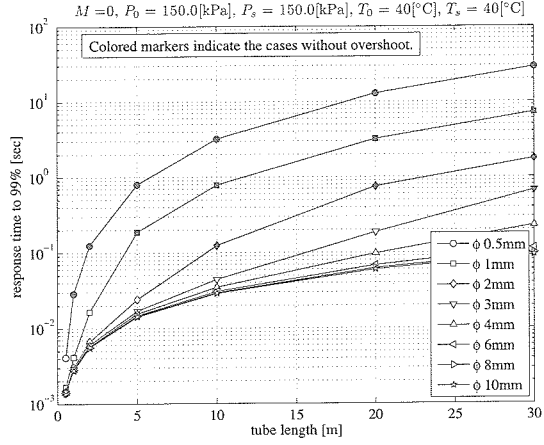
そこで、内径の候補を 2mm、4mm、6mm として、さらに絞込みを行うためにボード線図を作成した (図 12)。図中上半分は振幅の増加量から 1 を引いた数値である。また、風洞通風中の P_0 、 P_s 変動に相当する周波数を色付きの領域で表した。

これらの図 (図 10, 11, 12) を用いて、以下のような判断を行った。

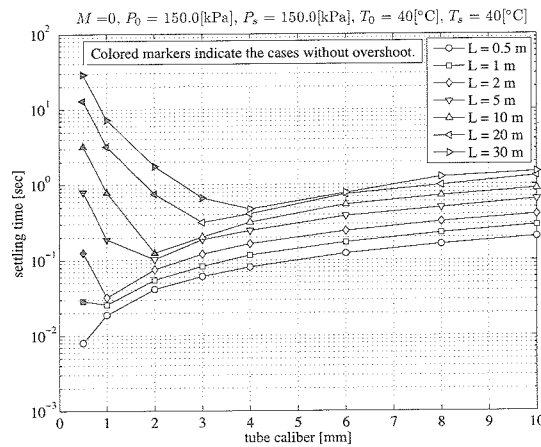
- 図 12 から分かるように、 P_0 および P_s の変動は非常に低い周波数が主体となっている。従って、図 12 から分かるように、配管を経由して計測することによるゲインの変化はまったく問題にならないし、位相のずれも問題になる程度では無い。
- 同様に、変動の主な周波数が非常に低いことから、オーバーシュートが問題になるようなことはない。また、 P_s を計測するプリナム室では風が舞っている状態になるが、静圧計測配管末端 (計測点) をパンチングメタルで囲った静圧計測箱の中に設置しているため、高い周波数帯域の圧力振動は取り除かれると考えられる。従って、オーバーシュートがあること自体は問題ではない。
- 図 12 より、いずれの配管でもゲインおよび位相のずれは問題になる程度では無いが、周囲圧力や温度の変化に対して、6mm の配管が最も特性の変化が少ない。
- 図 10(c), 11(c) から 2mm の配管では条件の変化でオーバーシュートの有無が切り替わることがわか



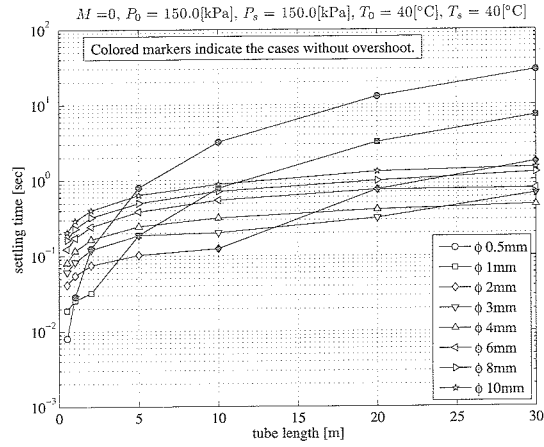
(a) response time と内径の関係.



(b) response time と管長の関係



(c) settling time と内径の関係.



(d) settling time と管長の関係.

図 11: 内径と管長を変えたときの応答チャート ($M = 0.0$, $P_0 = 150\text{kPa}$, $T_0 = 40^\circ\text{C}$).

る。また、4mm の配管でも $M = 1.4$ はオーバーシュートが起こらなくなる条件の境目である。基本的に、オーバーシュートは問題にならないが、通風条件によって特性に非線形な特性の変化が生じることは避けたい。

以上より、静圧変動という低周波で特定の周波数帯域に偏在した変動圧力を計測するという条件から、オーバーシュートを考慮する必要は無いと考えられ、配管長を 5m に設定することができたことで、いずれの配管内径候補でも周波数応答による計測値への影響は問題にならないことが分かった。

従って、候補になったいずれの内径の配管を使用しても問題は無かったと考えられるが、敢えて内径の選択をするにあたっては、風洞の設定総圧の全範囲にわたって特性変化の少ない内径 6mm の配管を選択することとした。

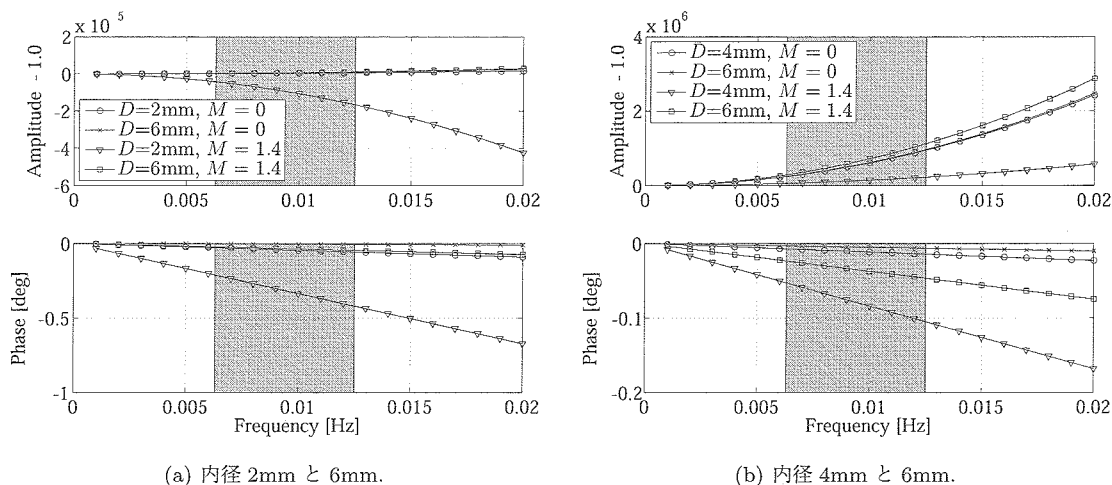


図 12: 配管候補のボード線図.

7 おわりに

宇宙航空研究開発機構 2m×2m 遷音速風洞では風洞基本量計測の高精度化のために、総圧 P_0 、静圧 P_∞ 計測用配管の見直しを行った。その過程で、連続風洞の定常運転で生じるような長周期の微小な圧力変動でも P_0 と P_∞ の計測に配管の圧力応答に起因する位相差が生じると、マッハ数算出に影響を及ぼすことが判明したため、 P_0 および P_∞ の計測配管を最適化して再敷設することにした。

まず、最適な配管の使用を決定するために文献 [1] を基にして配管の圧力応答特性を推算するプログラムを作成し、その妥当性を検証した。これを用いて解析を行うことによって配管の長さや内径を決定し、新規敷設配管の最適化を行うことができた。

8 謝辞

本研究の遂行に当たって風洞技術開発センター高度化セクションの口石茂研究員、栗田充研究員、藤井啓介リーダには有益なコメントをいただきました。ここに改めて感謝の意を表します。

A 作成したプログラムと使用方法

A.1 概要

作成したプログラムは数値処理システム MATLAB[®] のクラス drpms として実装した。ソースコードは Appendix C に示されている。使用するためには@drpms フォルダを含むフォルダを MATLAB[®] のサーチパスに追加する必要がある。

クラス drpms は公開メソッドとして

drpms	コンストラクタ
bode	ボード線図作成
step	ステップ応答作成
transient	任意入力に対する応答の作成
getresponse	圧力応答の計算 (bode, step, transient で使用する。)
set	パラメータの設定
get	パラメータの取得
plus	オブジェクトに対する加算演算子 (2 項+演算子) の再定義

を定義している。

公開メソッドの一覧は MATLAB[®] シェル上で以下のように参照することができる。

```
help drpms
```

また、使用方法は簡潔にソースコード内に記述されており、個々のメソッドの使用方法は以下のように参照することができる。

```
help drpms/bode
```

開発および動作確認は MATLAB[®] の version 7 上で行われた。

本節では、以下に入出力変数の構造および使用方法を詳細に述べる。ただし、MATLAB[®] については製品のマニュアルを参照のこと [5]。

A.2 使用方法

drpms は MATLAB[®] クラスになっているため、オブジェクト指向プログラミングに慣れていないユーザーのためにサンプルを使用しながら使用方法を解説する。詳細なプログラムの仕様についてはヘルプメッセージかソースコードを参照されたい。

A.2.1 配管が 1 本の場合

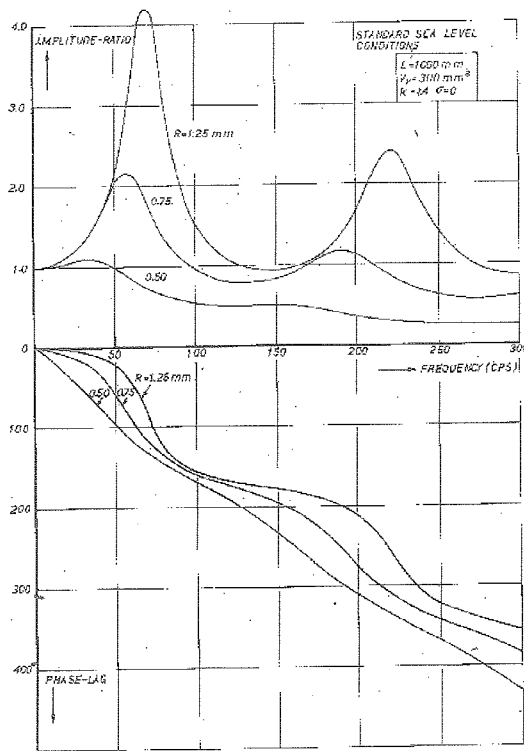
まず、系が 1 本の配管と一つのセンサーからなる場合として、文献 [1] の Fig. 10. を例題として取り上げる。

文献との結果の比較

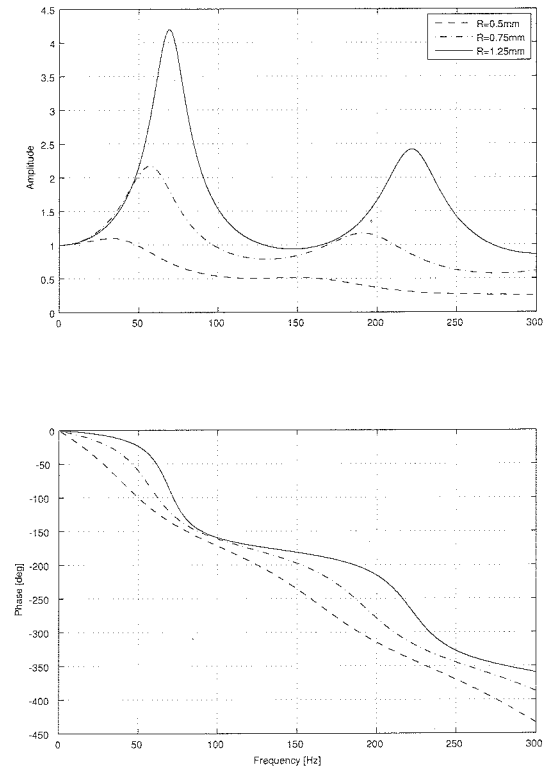
この例では長さ 1000mm の配管を、内部容積 300mm³ で圧力変動による容積変化率 0.0 である圧力センサーに接続した際のセンサーの圧力応答を 3 種類の配管内径 (半径 0.5mm、0.75mm、1.25mm) について計算している。

すなわち、図 1 において $L = 300\text{mm}$ 、 $V_v = 300\text{mm}^3$ 、 $R = 0.5, 0.75, 1.25\text{mm}$ とした場合である。

まず、文献の結果と drpms の出力を比較してみる。文献から引用したボード線図を図 13(a) に示し、drpms で作成したボード線図を図 13(b) に示す。両者の一致は良く、drpms は文献 [1] のボード線図を再現できることが分かる。図 13(b) では破線と点線を使用しているが、これはモノクロで出力するために線種を変更したためで、デフォルトではカラーの実線で出力される。



(a) 文献 [1] Fig.10.



(b) 図 14 で作成したボード線図.

図 13: 配管が 1 本の例.

Matlab[®] コードの解説

次に、この例のボード線図を作成する実際の MATLAB[®] スクリプトを図 14 に示す。

drpms はオブジェクト指向で設計されているので、クラスと称される雛形に必要なパラメータを設定して作成したオブジェクト、すなわち計算対象となる系を計算機上に再現した「もの」に対してボード線図の作成などの計算指示を出す、というのが基本的な流れである。

まず、その「もの (=オブジェクト)」を作成するために、必要なパラメータを指定しているのが 4 行目から 7 行目である。

今回は配管の長さが一定でその内径を変化させることによる応答の違いを見ているので

Caliber 変数	3 種類の配管の内径 (m 単位)	4 行目
Length 変数	配管の長さ (m 単位)	5 行目
Volume 変数	圧力トランスデューサの内容積 (m ³ 単位)	6 行目
Sigma 変数	圧力トランスデューサの圧力変動による容積変化率 (無次元)	7 行目

をそれぞれ設定する。単位系は MKS である。

9~11 行目で以上の変数を使用して drpms オブジェクト p を作成する。

```

1  clear all
2
3  %Fig 10.
4  Caliber = [.5 .75 1.25]*1e-3*2;%inner diameter of the tube
5  Length = 1000e-3;%tube length
6  Volume = 300*(1e-3)^3;%pressure transducer volume
7  Sigma = 0;%dimensionless increase in transducer volume
8  %
9  for num = 1:length(Caliber)
10     p(num) = drpms(Caliber(num), Length, Volume, Sigma);
11 end
12
13 Ts = 15*ones(length(p),1)+273.15;
14 Ps = 101325*ones(length(p),1);
15
16 fmat = [1 300];
17 bode(p, Ts, Ps, fmat);
18 subplot(2,1,1)
19 legend('R=0.5mm','R=0.75mm','R=1.25mm')
20 h = get(gca, 'children');
21 set(h(2), 'LineStyle', '-.')
22 set(h(3), 'LineStyle', '--')
23 h = get(subplot(2,1,2), 'children');
24 set(h(2), 'LineStyle', '-.')
25 set(h(3), 'LineStyle', '--')

```

図 14: 配管が 1 本の例を実行するスクリプト.

ボード線図の作成には周囲温度、周囲気圧、計算周波数が必要であるので、13 行目と 14 行目で周囲温度と周囲気圧をそれぞれ定義している。これらはそれぞれのオブジェクトに対して一つずつ設定する必要があるので、 p と同じ長さのベクトルとして定義している。計算周波数は周波数刻みと最大周波数で設定することになっており、16 行目で $fmat = [周波数刻み \text{ 最大周波数}]$ の形で定義されている。

上で作成した $drpms$ オブジェクト p に対してボード線図の作成を指示しているのが 17 行目である。

この例では単純にグラフの出力を行っただけであるが、グラフの作成に使用する数値データを得たい場合には 17 行目を

```
[mag, phse, f] = bode(p, Ts, Ps, fmat)
```

と入力すれば、ゲイン mag 、位相遅れ $phse$ 、計算した周波数（グラフの横軸） f がベクトルとして得られる。このように $bode$ の出力を要求した場合はグラフは作成されないので注意が必要である。

A.2.2 配管 2 本を接続した場合

次の例として配管 2 本を接続した場合を扱う。この解析では図 1 のような配管・センサー系が単純に接続されたものとして配管の接続を扱っている（図 15(a)）が、実際の系では途中にセンサーが挟まったような系はあまりなく、途中に挟まったセンサーの内容積を 0 とすれば（図 15(b)）、途中に気室が無いような

扱いはできる。

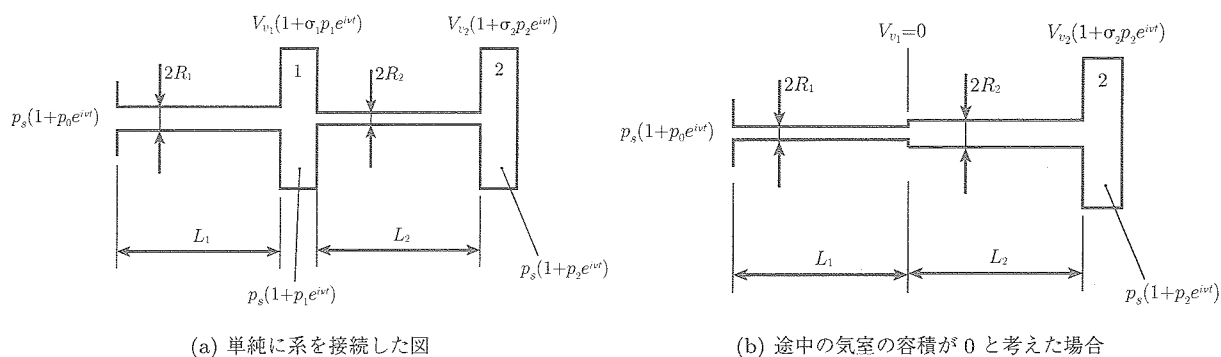


図 15: 配管が 2 本を接続した場合.

文献との結果の比較

ここでは文献 [1] の Fig. 16. を例題として取り上げる。

この例では配管内半径 1.5mm、長さ 250mm の配管に、長さ 750mm で 3 種類の内径を持つチューブを（配管内半径 0.5mm、0.75mm、1.25mm）を接続し、内部容積 300mm³ で圧力変動による容積変化率 0.0 である圧力センサーで計測した場合のセンサーの圧力応答を計算している。

すなわち、図 15(b) において $L_1 = 250\text{mm}$ 、 $V_{v1} = 0\text{mm}^3$ 、 $R_1 = 0.75\text{mm}$ 、 $L_2 = 750\text{mm}$ 、 $V_{v2} = 300\text{mm}^3$ 、 $R_2 = 0.5, 0.75, 1.25\text{mm}$ とした場合である。

この例でも、文献 [1] の結果と drpms の出力を比較してみる。文献から引用したボード線図を図 16(a) に示し、drpms で作成したボード線図を図 16(b) に示す。両者の一致は良く、drpms は文献 [1] のボード線図を再現できることが分かる。

Matlab® コードの解説

次に、この例のボード線図を作成する実際の MATLAB® スクリプトを図 17 に示す。ただし、図中の行番号は説明の便宜上付けられているもので、実際には入力してはならない。以降の例でも、行番号の扱いは同様である。

この例では二本の配管を接続しているので、4 行目から 10 行目まででそれぞれの配管のパラメータを定義している。

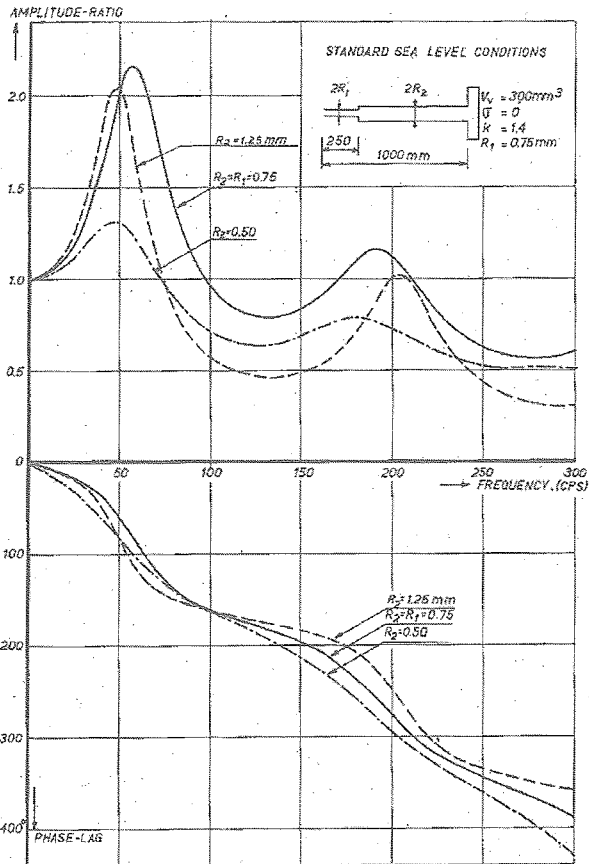
一本目の配管はケースごとにパラメータが変化しないので、12 行目で p1 として一回だけ定義する。二本目の配管はケースごとに内径が変化するため、13～16 行目の for ループで囲み、p2 として定義し、15 行目で p1 と接続する。

15 行目では以下のような加算演算子 (+) を使用した表現が登場するが、これは演算子を drpms オブジェクトに対して再定義したことによって使用できるようになっているもので、この演算は可換ではない ($p1 + p2 \neq p2 + p1$)。すなわち、この場合は p1 で定義された被演算子のほうが p2 よりも開放端側であることが要求される（図 15(b) では左側）。

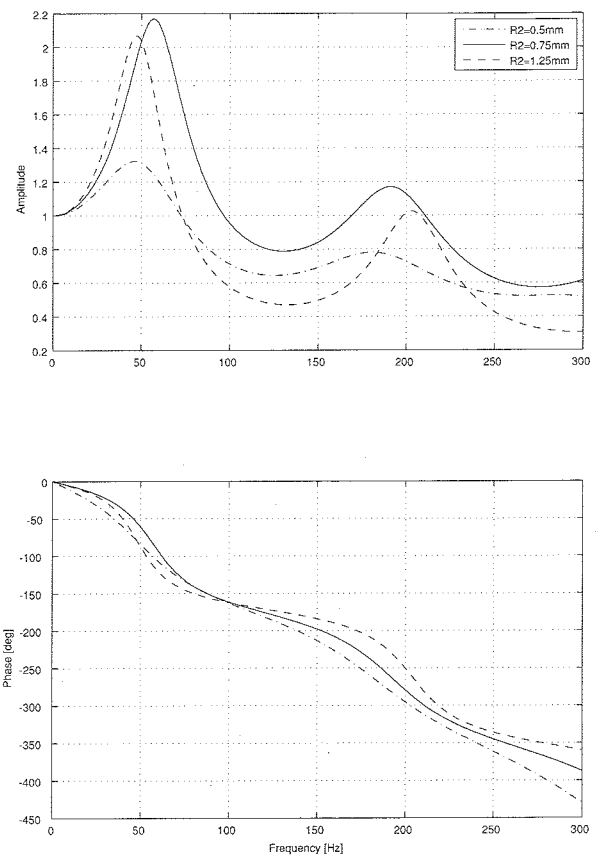
```
p(num) = p1 + p2;
```

同様に、以下のように記述した場合、p1 から p4 は開放端からセンサー側末端へ向かって順に接続されていなければならない。

```
p = p1 + p2 + p3 + p4;
```



(a) 文献 [1] Fig.16.



(b) 図 14 で作成したボード線図

図 16: 配管が 2 本の例.

ボード線図の作成 (18~22 行目) は図 14 の 13~17 行目とまったく同様であり、オブジェクトの作成をしてしまえば、配管が何本接続された系でも扱いは変わらない。

```

1  clear all
2
3  %Fig 16.
4  Caliber1 = .75e-3*2;%inner diameter of the tube1;
5  Caliber2 = [.5 .75 1.25]*1e-3*2;%inner diameter of the tube2
6  Length1 = 250e-3;%tube1 length
7  Length2 = 750e-3;%tube2 length
8  Volume1 = 0.0;%pressure transducer1 volume
9  Volume2 = 300*(1e-3)^3;%pressure transducer2 volume
10 Sigma = 0;%dimensionless increase in transducer volume
11 %
12 p1 = drpms(Caliber1, Length1, Volume1, Sigma);%tube1 construction
13 for num = 1:length(Caliber2)
14     p2 = drpms(Caliber2(num), Length2, Volume2, Sigma);%tube2 construction
15     p(num) = p1 + p2;%tube1 & tube2 connection
16 end
17
18 Ts = 15*ones(length(p),1)+273.15;
19 Ps = 101325*ones(length(p),1);
20
21 fmat = [1 300];
22 bode(p, Ts, Ps, fmat);
23 subplot(2,1,1)
24 legend('R2=0.5mm','R2=0.75mm','R2=1.25mm')
25 h = get(gca, 'children');
26 set(h(3), 'LineStyle', '-.')
27 set(h(1), 'LineStyle', '--')
28 h = get(subplot(2,1,2), 'children');
29 set(h(3), 'LineStyle', '-.')
30 set(h(1), 'LineStyle', '--')

```

図 17: 配管が2本の例を実行するスクリプト.

接続の取り扱い 接続された配管を扱う場合の注意として、接続の際にニューマチック・コネクタなどを使用して狭窄部が設けられている場合、5.2 で述べられているように影響を無視できない場合が多いので、これを一つの細い配管としてシミュレーションに組み入れることが必要である。

A.2.3 パラメータの変更

これまでの例では空気を作動気体とし、気室内での気体の膨張は等エントロピー的であるとして計算を行ってきた。drpms は特に指定しない限り地球上で空気を作動気体とすることを前提としているため、表 3 に記載したデフォルト値を使用して計算しているが、これらの数値は変更することができる。3.2 節で記述した条件を満足する限りにおいて、変更をしても妥当な結果を得られることが期待できる。(ただし、実験的な検証は行われていないことに注意が必要である。)

変更は公開メソッド set で行う。

変数名	変数の意味	デフォルト値
g	重力加速度 $g[\text{m/s}^2]$	9.80665
lambdaTable	作動気体の熱伝導率 $\lambda[\text{W}\cdot\text{m}^{-1}\cdot\text{K}^{-1}]$ を決定する数表	(*1)
gamma	比熱比 γ	1.4
k	ポリトロップ指数	1.4
R0	気体定数	287
Ts0	サザランドの公式における係数を定義する温度 [K]	293.15
mu0	Ts0[K]、1atm での粘性係数 $[\text{Pa}\cdot\text{s}]$	18.2e-6
S	Ts0[K]、1atm でのサザランドの定数	117
(*1) private メソッド airconductivity で決定し、 1 列目を絶対温度、2 列目を熱伝導率とする行列		

表 3: 変更可能なパラメータ.

ここでは、set の使用法の例として、重力加速度を地上の 1/6、作動気体を水素として前項の例を計算してみる。周囲大気圧は 101325Pa、周囲気温は 15°C とする。これらの条件はあくまでパラメータの変更方法を提示するために設定したものであり、物理的な意味づけは無い。出力されたボード線図は図 18 に示す。

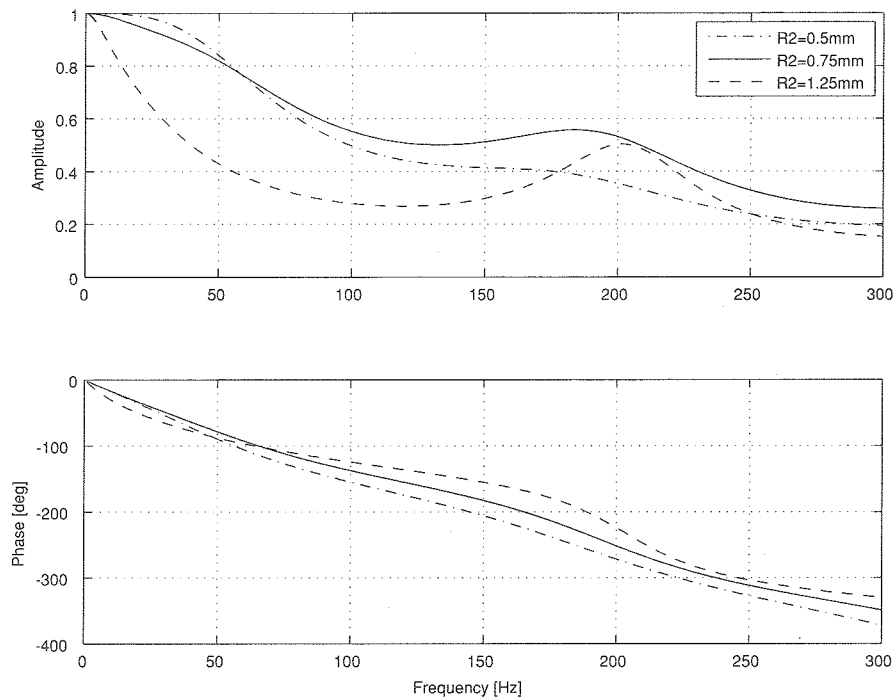


図 18: 文献 [1] の Fig. 16. の例で重力加速度を地上の 1/6、作動気体を水素として描いたボード線図.

Matlab[®] コードの解説

この場合の MATLAB[®] スクリプトを図 19 に示す。16～25 行目で各パラメータを変更している。set の文法は以下のものであり、第 1 引数として drpms オブジェクトを指定し、第 2 引数以降に変更可能なパラメータ名とその値を組として入力する。

```
p(num) = set(p(num), 'g', 9.80665/6);
```

パラメータ名／値の組を複数個連続して入力することもできる。

```
p(num) = set(p(num), 'g', 9.80665/6, 'R0', 4124);
```

ここで、注意しなければならないことは MATLAB[®] にはポインタや参照は無いため、値を変更した場合、set の出力を元のオブジェクトに代入しなければならない点である。特に、グラフィックハンドルなどの MATLAB[®] に元から存在するオブジェクトではこのような代入は必要ないため、気をつけなければならない。

変更可能なパラメータはコマンドラインで以下のように入力して表示できる。>>はコマンドプロンプトで、2 行目以降が出力である。

```
>> get(p(1))  
g : 1.6344  
lambdaTable :  
Temperature : 7.3150e+001 1.7315e+002 2.7315e+002 3.7315e+002  
Conductivity : 5.0900e-002 1.1240e-001 1.6820e-001 2.1180e-001  
gamma : 1.4  
k : 1.4  
R0 : 4124  
mu0 : 8.8e-006  
S : 72
```



```

1  clear all
2
3  %Fig 16.
4  Caliber1 = .75e-3*2;%inner diameter of the tube1;
5  Caliber2 = [.5 .75 1.25]*1e-3*2;%inner diameter of the tube2
6  Length1 = 250e-3;%tube1 length
7  Length2 = 750e-3;%tube2 length
8  Volume1 = 0.0;%pressure transducer1 volume
9  Volume2 = 300*(1e-3)^3;%pressure transducer2 volume
10 Sigma = 0;%dimensionless increase in transducer volume
11 %
12 p1 = drpms(Caliber1, Length1, Volume1, Sigma);%tube1 construction
13 for num = 1:length(Caliber2)
14     p2 = drpms(Caliber2(num), Length2, Volume2, Sigma);%tube2 construction
15     p(num) = p1 + p2;%tube1 & tube2 connection
16     %parameter changes
17     T = [-200; -100; 0; 100]+273.15;
18     lambda = [5.09; 11.24; 16.82; 21.18]*1e-2;
19     p(num) = set(p(num), ...
20         'g', 9.80665/6, ...%gravity is 1/6G.
21         'lambdaTable', [T lambda], ...%thermal conductivity of H2
22         'gamma', 1.4, ...%specific heat ratio of H2
23         'R0', 4124, ...%gas constant of H2
24         'mu0', 8.8e-6, ...%viscosity of H2 at Ts0 and 1atm
25         'S', 72);%Sutherland's constant of H2 at 1atm and p.Ts0
26 end
27
28 Ts = 15*ones(length(p),1)+273.15;
29 Ps = 101325*ones(length(p),1);
30
31 fmat = [1 300];
32 bode(p, Ts, Ps, fmat);
33 subplot(2,1,1)
34 legend('R2=0.5mm','R2=0.75mm','R2=1.25mm')
35 h = get(gca, 'children');
36 set(h(3), 'LineStyle', '-.')
37 set(h(1), 'LineStyle', '--')
38 h = get(subplot(2,1,2), 'children');
39 set(h(3), 'LineStyle', '-.')
40 set(h(1), 'LineStyle', '--')

```

図 19: パラメータの変更の例.

A.2.4 ステップ応答

周波数応答（ボード線図）は非定常計測を行うときに有効に使えるが、配管系を設計する場合には単純にどの程度応答が遅れるのかのイメージを掴みたいときも多く、このような場合はステップ応答が有用である。

そこで、drpms クラスにはステップ応答を算出する機能を用意した。プログラムの実装としては、配管の周波数応答特性はすでに得られているので、これにステップ入力をディジタルフーリエ変換したものを掛けて、結果を逆フーリエ変換することによってステップ応答を算出している。ただし、ここで得られるステップ応答は微小変動を仮定した周波数応答特性から算出されており、大圧力差の入力に対しては前提が崩れてしまうので、そのまま使用することは適切ではない。

ステップ応答のグラフを得るコマンドはボード線図を計算するときと同様に簡潔であり、図 14 の 17 行目以降を以下のコマンドに差し替えればステップ応答のグラフが得られる。(本質的にはこのコード断片の 1 行目のみが必要)

図 20 から、太い配管 ($R=1.25\text{mm}$) ではステップ入力によって発生した衝撃波が管端で反射して往復するのが観察されるが、細い配管では管内の気体の粘性によってすぐに減衰してしまうことが分かる。

```
step(p, Ts, Ps, fmat)
legend('R=0.5mm', 'R=0.75mm', 'R=1.25mm')
h = get(gca, 'children');
set(h(2), 'LineStyle', '-.')
set(h(3), 'LineStyle', '--')
```

ボード線図のときと同様に、以下のコマンドでステップ応答図を作成するための数値データを得ることができる。このときも図の作成は抑制される。

```
[y, t] = step(p, Ts, Ps, fmat)
```

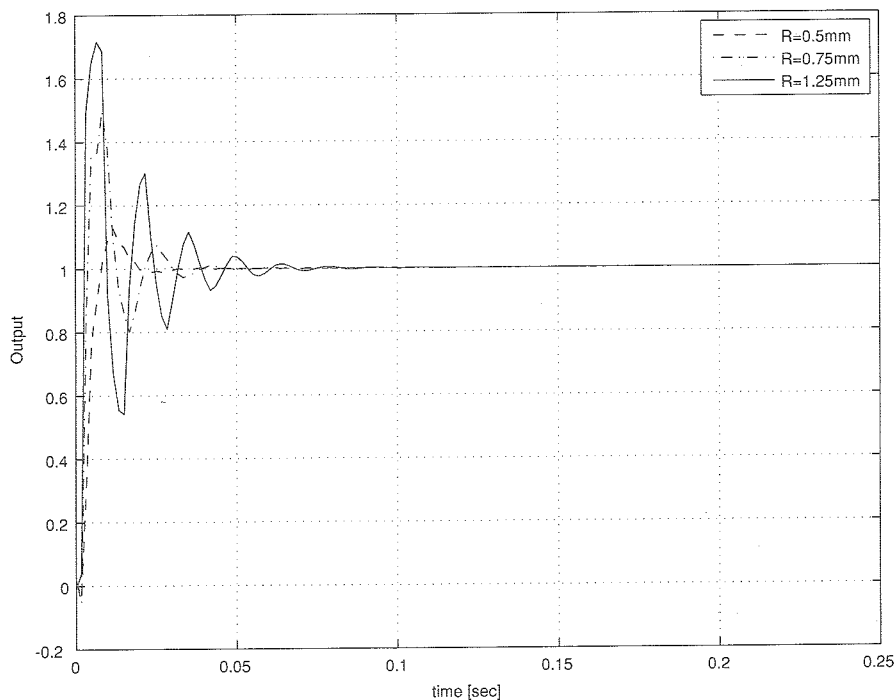


図 20: A.2.1 節の例におけるステップ応答の計算例.

A.2.5 任意入力に対する応答

ステップ応答を求めるのと同様の方法をとれば 0 から 1 に変化する任意の入力に対して配管の応答を求めることができる。そこで、任意入力に対する応答を計算するメソッド `transient` を用意した。`step` の実装は実際には `transient` を利用してステップ応答を計算している。

このメソッドはもともと `step` のヘルパ関数として用意されたもので、`bode` や `step` とは呼び出し方が少し異なる。図 21 に A.2.1 節の例に前の A.2.4 で与えたステップ入力の変わりに 0.01 秒で線形に立ち上がる入力を与えた場合のコードの例を示す。ここではこれまでの周波数だけを与える呼び出し方法とは異なり、

```
x0 = [(0:0.01:1)']; ones(400,1)];
tmax = 0.05;
transient(p, Ts, Ps, x0, tmax)
```

というように、実際の入力波形とその継続時間を与える。

このときの出力は図 22 のようになり、図の上半分が入力、下半分がその入力を与えられた場合の応答を示す。入力の高周波成分がカットされているため、ステップ応答のときのような鋭い立ち上がりは観測されない。実際の場面では鋭い立ち上がりの正確なステップ入力が増加されることはまれであり、この機能は実際的な入力に対する応答を確認する場面で役立つことが期待される。

```
1 clear all
2
3 Caliber = [.5 .75 1.25]*1e-3*2;%inner diameter of the tube
4 Length = 1000e-3;%tube length
5 Volume = 300*(1e-3)^3;%pressure transducer volume
6 Sigma = 0;%dimensionless increase in transducer volume
7 %
8 for num = 1:length(Caliber)
9     p(num) = drpms(Caliber(num), Length, Volume, Sigma);
10 end
11
12 Ts = 15*ones(length(p),1)+273.15;
13 Ps = 101325*ones(length(p),1);
14
15 x0 = [(0:0.01:1)']; ones(400,1)];
16 tmax = 0.05;
17 transient(p, Ts, Ps, x0, tmax)
18 subplot(2,1,2)
19 legend('R=0.5mm','R=0.75mm','R=1.25mm')
20 h = get(gca, 'children');
21 set(h(2), 'LineStyle', '-.')
22 set(h(3), 'LineStyle', '--')
23 h = get(subplot(2,1,2), 'children');
24 set(h(2), 'LineStyle', '-.')
25 set(h(3), 'LineStyle', '--')
```

図 21: 任意入力に対する応答の例.

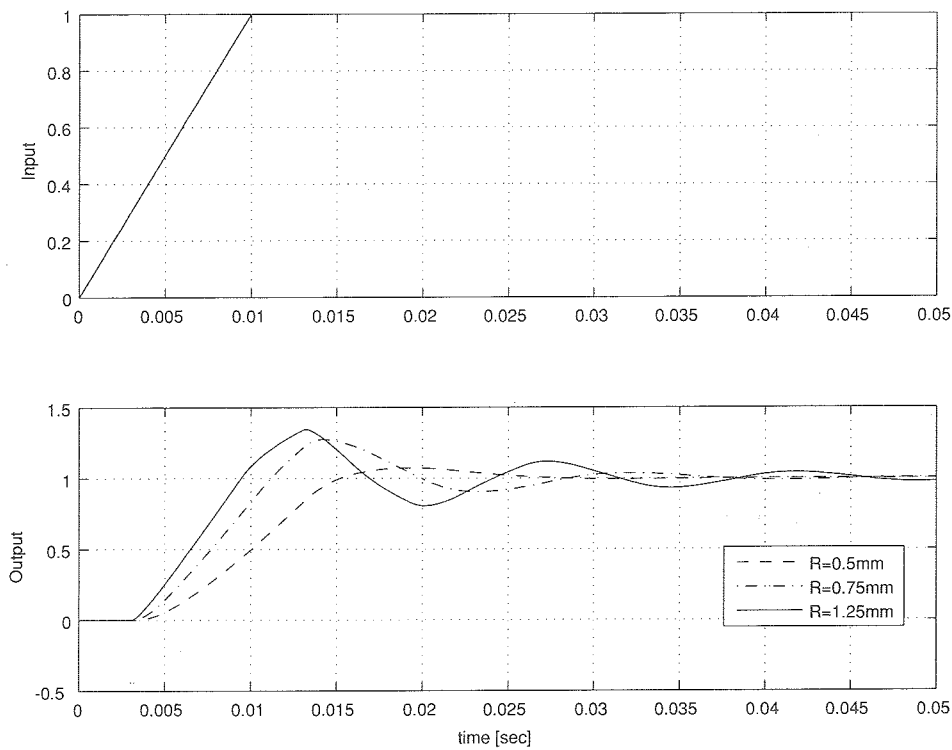


図 22: 任意入力に対する応答の計算例.

B 使用上の注意

本稿で作成されたプログラムには適用できる事例に制約があることを注記しておく。Tijdeман は風洞試験模型翼表面の圧力孔のように圧力孔が流れにさらされているときは、圧力応答が無風のときと異なることを指摘している [4, 2]。ここで紹介したプログラムはこの効果を入れておらず、そのような圧力孔の応答を推算するために使用することはできない。

C ソースコード

本節ではソースコードを添付する。ソースコードは以下のディレクトリ構成を保持した上で MATLAB[®] からパスの通ったフォルダに格納される必要がある。

- @drpms

drpms.m コンストラクタ

get.m パラメータ取得

set.m パラメータ設定

getresponse.m 圧力応答計算

bode.m ボード線図作成

step.m ステップ応答線図作成

transient.m 任意入力応答線図作成

plus.m 加算演算子の再定義

- @drpms/private

viscosity.m 粘性係数計算

airconductivity.m 空気の熱伝導率を出力する（数値データは [6] による）

@drpms には公開メソッドが格納され、drpms オブジェクトに対して使用することができる。@drpms/private の中身は公開メソッドによって利用される非公開メソッドであり、ユーザーが直接呼び出して使用することはできない。

C.1 drpms.m

```

1  function p = drpms(caliber, len, volume, sigma, nextSystem)
2  %
3  %   DRPMS is the constructor of drpms (dynamic response of pressure measuring systems) class.
4  %
5  %   P = DRPMS(CALIBER, LEN, VOLUME, SIGMA) returns an object of the drpms class.
6  %   The object implements the series connection of tubes and transducers.
7  %
8  %   CALIBER is the inner diameter of the tube (in meter), LEN is the length
9  %   of the tube (in meter), VOLUME is the pressure transducer volume and SIGMA is
10 %   a dimensionless increase in transducer volume due to diaphragm deflection.
11 %
12 %   P = DRPMS(CALIBER, LEN, VOLUME, SIGMA, NEXTSYSTEM) returns an object
13 %   which represent the tube & pressure transducer system adjacent to NEXTSYSTEM.
14 %
15 %   This class design is based on NLR-TR F.238.
16 %
17 %   Example:
18 %
19 %   This example represents the system which were examined in Fig 16. of NLR-TR F.238
20 %
21 %   %Fig 16.
22 %   Caliber1 = .75e-3*2;%inner diameter of the tube1;
23 %   Caliber2 = [.5 .75 1.25]*1e-3*2;%inner diameter of the tube2
24 %   Length1 = 250e-3;%tube1 length

```

```

25 % Length2 = 750e-3;%tube2 length
26 % Volume1 = 0.0;%pressure transducer1 volume
27 % Volume2 = 300*(1e-3)^3;%pressure transducer2 volume
28 % Sigma = 0;%dimensionless increase in transducer volume
29 % %
30 % p1 = drpms(Caliber1, Length1, Volume1, Sigma);%tube1 construction
31 % for num = 1:length(Caliber2)
32 %     p2 = drpms(Caliber2(num), Length2, Volume2, Sigma);%tube2 construction
33 %     p(num) = p1 + p2;%tube1 & tube2 connection
34 % end
35 %
36 % Ts = 15*ones(length(p),1)+273.15;
37 % Ps = 101325*ones(length(p),1);
38 %
39 % fmat = [1 300];
40 % bode(p, Ts, Ps, fmat);
41 %
42 % See also drpms/get, drpms/set, drpms/getresponse, drpms/bode, drpms/step,
43 % drpms/transient, drpms/plus
44 %
45 % Author:
46 % Makoto Ueno (ueno.makoto@jaxa.jp)
47 %
48 % License:
49 %
50 % Copyright (c) 2006, Japan Aerospace Exploration Agency
51 % All rights reserved.
52 %
53 % Redistribution and use in source and binary forms, with or without modification,
54 % are permitted provided that the following conditions are met:
55 % 1. Redistributions of source code must retain the above copyright notice,
56 %    this list of conditions and the following disclaimer.
57 % 2. Redistributions in binary form must reproduce the above copyright notice,
58 %    this list of conditions and the following disclaimer in the documentation
59 %    and/or other materials provided with the distribution.
60 % 3. Neither the name of the Japan Aerospace Exploration Agency nor the names
61 %    of its contributors may be used to endorse or promote products derived from
62 %    this software without specific prior written permission.
63 %
64 % THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
65 % ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
66 % WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
67 % DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
68 % ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
69 % (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
70 % LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
71 % ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
72 % (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
73 % SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
74 %
75 p.g = 9.80665;%acceleration of gravity
76 p.lambdaTable = airconductivity;%thermal conductivity table
77 p.gamma = 1.4;%specific heat ratio
78 p.k = 1.4;%polytropic exponent
79 p.R0 = 287;%gas constant of air
80 p.Ts0 = 293.15;%Temperature of air at which the Sutherland's constant is defines.

```

```

81 p.mu0 = 18.2e-6;%viscosity of air at 1atm and p.Ts0
82 p.S = 117;%Sutherland's constant of air at 1atm and p.Ts0
83
84 if nargin==0
85     p.tube.radius = [];
86     p.tube.length = [];
87     p.tube.volume = [];
88     p.volume.sigma = [];
89     p.volume.volume = [];
90     p.nextSystem = [];
91 else
92     error(nargchk(4, 5, nargin))
93     if ~(isnumeric(caliber)&isnumeric(len)&isnumeric(volume)&isnumeric(sigma))
94         error('@drpms/drpms: input args must be numeric.')
95     elseif ~(length(caliber)==1&length(len)==1&length(volume)==1&length(sigma)==1)
96         error('@drpms/drpms: input args must have length of 1.')
97     end
98     if nargin == 4
99         p.tube.radius = caliber/2;
100        p.tube.length = len;
101        p.tube.volume = pi.*p.tube.radius.^2.*p.tube.length;
102        p.volume.sigma = sigma;
103        p.volume.volume = volume;
104        p.nextSystem = [];
105    elseif nargin == 5
106        p.tube.radius = caliber/2;
107        p.tube.length = len;
108        p.tube.volume = pi.*p.tube.radius.^2.*p.tube.length;
109        p.volume.sigma = sigma;
110        p.volume.volume = volume;
111        p.nextSystem = nextSystem;
112    end
113 end
114 p = class(p, 'drpms');

```

C.2 get.m

```

1 function value = get(p, name)
2 % GET returns the attribute of DRPMS object.
3 %
4 % GET(P) returns the value of all of the changeable attributes of DRPMS object.
5 %
6 % VALUE = GET(P, NAME) returns the attribute of DRPMS object.
7 % You can specify the attribute name, NAME.
8 % NAME must be 'g', 'lambdaTable', 'gamma', 'k', 'RO', 'mu0' or 'S'.
9 %     'g'      -- acceleration of gravity
10 %     'lambdaTable' -- thermal conductivity table
11 %     'gamma'  -- specific heat ratio
12 %     'k'      -- polytropic exponent
13 %     'RO'     -- gas constant
14 %     'Ts0'    -- Temperature of air at which the Sutherland's constant is defines
15 %     'mu0'    -- viscosity of gas at 1atm and 293.15K(=20.0deg.C)
16 %     'S'      -- Sutherland's constant of gas at 1atm and 293.15K(=20.0deg.C)
17 %     'nextSystem' -- The next system
18 %

```

```

19 % See also drpms, drpms/set, drpms/getresponse, drpms/bode, drpms/step,
20 % drpms/transient, drpms/plus
21
22 % Author:
23 % Makoto Ueno (ueno.makoto@jaxa.jp)
24 %
25 % License:
26 %
27 % Copyright (c) 2006, Japan Aerospace Exploration Agency
28 % All rights reserved.
29 %
30 % Redistribution and use in source and binary forms, with or without modification,
31 % are permitted provided that the following conditions are met:
32 % 1. Redistributions of source code must retain the above copyright notice,
33 % this list of conditions and the following disclaimer.
34 % 2. Redistributions in binary form must reproduce the above copyright notice,
35 % this list of conditions and the following disclaimer in the documentation
36 % and/or other materials provided with the distribution.
37 % 3. Neither the name of the Japan Aerospace Exploration Agency nor the names
38 % of its contributors may be used to endorse or promote products derived from
39 % this software without specific prior written permission.
40 %
41 % THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
42 % ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
43 % WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
44 % DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
45 % ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
46 % (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
47 % LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
48 % ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
49 % (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
50 % SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
51
52     if length(p)~=1
53         error('@drpms/get: the length of ''p'' must be one.');
```

```

54     end
55     varnames = {'g', 'lambdaTable', 'gamma', 'k', 'R0', 'Ts0', 'mu0', 'S', 'nextSystem'};
56
57     if ~exist('name','var')
58         fn = varnames;
59         for num = 1:length(fn)
60             if strcmp(fn{num}, 'lambdaTable')
61                 fprintf(1, [fn{num} ' ' : '\n'])
62                 str = ['\tTemperature : ' ...
63                     num2str(p.(fn{num})(: ,1), '%5.4e ') '\n'];
64                 fprintf(1, str);
65                 str = ['\tConductivity : ' ...
66                     num2str(p.(fn{num})(: ,2), '%5.4e ') '\n'];
67                 fprintf(1, str);
68             elseif strcmp(fn{num}, 'nextSystem')
69                 if isempty(p.(fn{num}))
70                     fprintf(1, [fn{num} ' ' : empty'\n'])
71                 else
72                     fprintf(1, [fn{num} ' ' : exist'\n'])
73                 end
74             else

```



```

75             fprintf(1, [fn{num} '\t : ' num2str(p.(fn{num})) '\n'])
76         end
77     end
78     else
79         switch name
80         case varnames
81             value = p.(name);
82         otherwise
83             error([''' name ''' is not a valid drpms property\n'...
84                 'drpms properties: g, lambdaTable, gamma, k, '...
85                 'R0, mu0, S, nextSystem'], [])
86         end
87     end
88

```

C.3 set.m

```

1  function p = set(p, varargin)
2  % SET returns the attribute of DRPMS object.
3  %
4  % P = SET(P, 'PropertyName1', PropertyValue1) sets the attribute of DRPMS.
5  % You can specify the attribute name, NAME.
6  % NAME have to be 'g', 'lambdaTable', 'gamma', 'k', 'R0', 'mu0' or 'S'.
7  %     'g'      -- acceleration of gravity
8  %     'lambdaTable' -- thermal conductivity table
9  %     'gamma'  -- specific heat ratio
10 %     'k'      -- polytropic exponent
11 %     'R0'     -- gas constant
12 %     'Ts0'    -- Temperature of air at which the Sutherland's constant is defines
13 %     'mu0'    -- viscosity of gas at 1atm and 'Ts0'
14 %     'S'      -- Sutherland's constant of gas at 1atm and 'Ts0'
15 %     'nextSystem' -- The next system
16 %
17 % P = SET(P, ''PropertyName1',PropertyValue1,'PropertyName2',PropertyValue2,...)
18 % sets multiple property values with a single statement. Note that it
19 % is permissible to use property/value pairs.
20 %
21 % See also drpms, drpms/get, drpms/getresponse, drpms/bode, drpms/step,
22 % drpms/transient, drpms/plus
23 %
24 % Author:
25 % Makoto Ueno (ueno.makoto@jaxa.jp)
26 %
27 % License:
28 %
29 % Copyright (c) 2006, Japan Aerospace Exploration Agency
30 % All rights reserved.
31 %
32 % Redistribution and use in source and binary forms, with or without modification,
33 % are permitted provided that the following conditions are met:
34 % 1. Redistributions of source code must retain the above copyright notice,
35 %    this list of conditions and the following disclaimer.
36 % 2. Redistributions in binary form must reproduce the above copyright notice,
37 %    this list of conditions and the following disclaimer in the documentation
38 %    and/or other materials provided with the distribution.

```

```

39 % 3. Neither the name of the Japan Aerospace Exploration Agency nor the names
40 %   of its contributors may be used to endorse or promote products derived from
41 %   this software without specific prior written permission.
42 %
43 % THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
44 % ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
45 % WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
46 % DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
47 % ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
48 % (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
49 % LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
50 % ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
51 % (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
52 % SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
53
54     if length(p)~=1
55         error('@drpms/set: the length of ''p'' must be one.');
```

```

56     end
57     propertyArgin = varargin;
58
59     if mod(length(propertyArgin), 2) ~= 0
60         error(['@drpms/set: you can use ''PeopertyName'' and '...
61             ''PropertyValue'' in pair.']);
62     end
63     while length(propertyArgin) >= 2
64         prop = propertyArgin{1};
65         val = propertyArgin{2};
66         if ~ischar(prop)
67             error(['@drpms/set: you can use ''PeopertyName'' and '...
68                 ''PropertyValue'' in pair.']);
69         else
70             switch prop
71                 case {'g', 'lambdaTable', 'gamma', 'k', 'RO', 'Ts0', 'mu0', 'S'}
72                     p.(prop) = val;
73                     if ~isempty(p.nextSystem)
74                         set(p.nextSystem, varargin{:});
75                     end
76                 case 'nextSystem'
77                     if isa(val, 'drpms')
78                         p.nextSystem = val;
79                     else
80                         error('''nextSystem'' have to be ''drpms'' object.');
```

```

81                     end
82                 otherwise
83                     error(['drpms properties: g, lambdaTable, ' ...
84                         'gamma, k, RO, Ts0, mu0, S'])
85             end
86         end
87         propertyArgin(1:2) = [];
88     end

```

C.4 getresponse.m

```

1 function plp0 = getresponse(p, Ts, Ps, f)
2 % GETRESPONSE returns the pressure output/intput ratio in complex numbers.

```

```

3 %
4 %   P1P0 = GETRESPONSE(P, TS, PS, F) returns the pressure output/input ratio
5 %   in the form of complex number array. The length of P, TS and PS must be 1.
6 %   P is the drpms object, TS is the ambient static temperature, PS is the ambient
7 %   static pressure and F is the frequency to be computed.
8 %   The response P1P0 is computed along with the frequency vector F.
9 %
10 %   See also drpms, drpms/get, drpms/set, drpms/bode, drpms/step,
11 %   drpms/transient, drpms/plus
12
13 %   Author:
14 %   Makoto Ueno (ueno.makoto@jaxa.jp)
15 %
16 %   License:
17 %
18 %   Copyright (c) 2006, Japan Aerospace Exploration Agency
19 %   All rights reserved.
20 %
21 %   Redistribution and use in source and binary forms, with or without modification,
22 %   are permitted provided that the following conditions are met:
23 %   1. Redistributions of source code must retain the above copyright notice,
24 %       this list of conditions and the following disclaimer.
25 %   2. Redistributions in binary form must reproduce the above copyright notice,
26 %       this list of conditions and the following disclaimer in the documentation
27 %       and/or other materials provided with the distribution.
28 %   3. Neither the name of the Japan Aerospace Exploration Agency nor the names
29 %       of its contributors may be used to endorse or promote products derived from
30 %       this software without specific prior written permission.
31 %
32 %   THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
33 %   ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
34 %   WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
35 %   DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
36 %   ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
37 %   (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
38 %   LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
39 %   ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
40 %   (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
41 %   SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
42
43 vars = {'p', 'Ts', 'Ps'};
44 for num = 1:length(vars)
45     if eval(['length(' vars{num} ')~=1'])
46         error(['@tjdemman/getresponse: the length of ''' vars{num} ...
47             ''' must be one.']);
48     end
49 end
50 s = size(f);
51 if length(s)~=2|all(s~=1)
52     error('@tjdemman/getresponse: ''f'' must be a vector.')
53 else
54     if s(1)==1
55         f = f';
56     end
57 end
58

```

```

59 [alpha{1}, x{1}, n{1}, phi{1}] = values(p, Ts, Ps, f);
60
61 if isempty(p.nextSystem)
62     p1p0 = ...
63         (cosh(phi{1}.*p.tube.length)+ ...
64         n{1}.*p.volume.volume./p.tube.volume.*(p.volume.sigma+1/p.k) ...
65         .*phi{1}.*p.tube.length.*sinh(phi{1}.*p.tube.length)).^(-1);
66 else
67     q = p.nextSystem;
68     [alpha{2}, x{2}, n{2}, phi{2}] = values(q, Ts, Ps, f);
69     p2p1 = getresponse(q, Ts, Ps, f);
70     p1p0 = ...
71         (cosh(phi{1}.*p.tube.length)+ ...
72         n{1}.*p.volume.volume./p.tube.volume.*(p.volume.sigma+1/p.k) ...
73         .*phi{1}.*p.tube.length.*sinh(phi{1}.*p.tube.length)+ ...
74         (q.tube.volume.*phi{1}.*p.tube.length.*n{1}.* ...
75         sinh(phi{1}.*p.tube.length))./ ...
76         (p.tube.volume.*phi{2}.*q.tube.length.*n{2}.* ...
77         sinh(phi{2}.*q.tube.length)).* ...
78         (cosh(phi{2}.*q.tube.length)-p2p1)).^(-1).*p2p1;
79 end
80 I = find(f==0);
81 p1p0(I) = 1;
82
83 function [Alpha, X, N, Phi] = values(p, Ts, Ps, f)
84
85 a0 = sqrt(p.gamma.*p.R0.*Ts);%velocity of sound
86 rho = Ps./(p.R0.*Ts);%density
87 mu = viscosity(Ts, p.Ts0, p.mu0, p.S);%viscosity
88 Cp = p.R0*p.gamma/(p.gamma-1);%speciic heat at constant pressure
89 lambda = interp1(p.lambdaTable(:,1), p.lambdaTable(:,2), Ts);%thermal conductivity
90 Pr = mu.*Cp./lambda;%Prandtl number
91 nu = 2*pi*f;%frequency (rad/sec)
92
93 Alpha = i^(3/2).*p.tube.radius.*sqrt((rho.*nu)./mu);
94 X = Alpha.*sqrt(Pr);
95 N = (1+(p.gamma-1)/p.gamma*besselj(2,X)./besselj(0,X)).^(-1);
96 Phi = nu./a0.*sqrt(besselj(0,Alpha)./besselj(2,Alpha).*p.gamma./N);

```

C.5 bode.m

```

1 function [mag, phse, f] = bode(p, Ts, Ps, fmat, yscale)
2 % BODE Bode frequency response of DRPMS object.
3 %
4 % BODE(P, TS, PS, [FRQ_STEP MAX_FRQ], YSCALE) draws the Bode plot of
5 % the DRPMS object P.
6 % The frequency range and number of points are based on a vector
7 % F = FRQ_STEP:FRQ_STEP:MAX_FRQ.
8 % The scale of the y-axis of the magnitude YSCALE is optional.
9 % You can specify the kind of the yscale from 'linear', 'db' or 'log'.
10 % The default is 'linear'.
11 %     'linear' -- linear scale.
12 %     'db'     -- decibel scale.
13 %     'log'    -- logarithmic scale.
14 %

```

```

15 % [MAG,PHSE,F] = BODE(P, TS, PS, [FRQ_STEP MAX_FRQ], YSCALE) return the
16 % response magnitudes and phases in degrees (along with the frequency
17 % vector F). No plot is drawn on the screen. If the length of P is N,
18 % MAG and PHSE are arrays of size [1+length(FMAT) N] where MAG(:,j) gives
19 % the response magnitudes of the j-th object.
20 %
21 % See also drpms, drpms/get, drpms/set, drpms/getresponse, drpms/step,
22 % drpms/transient, drpms/plus
23 %
24 % Author:
25 % Makoto Ueno (ueno.makoto@jaxa.jp)
26 %
27 % License:
28 %
29 % Copyright (c) 2006, Japan Aerospace Exploration Agency
30 % All rights reserved.
31 %
32 % Redistribution and use in source and binary forms, with or without modification,
33 % are permitted provided that the following conditions are met:
34 % 1. Redistributions of source code must retain the above copyright notice,
35 %    this list of conditions and the following disclaimer.
36 % 2. Redistributions in binary form must reproduce the above copyright notice,
37 %    this list of conditions and the following disclaimer in the documentation
38 %    and/or other materials provided with the distribution.
39 % 3. Neither the name of the Japan Aerospace Exploration Agency nor the names
40 %    of its contributors may be used to endorse or promote products derived from
41 %    this software without specific prior written permission.
42 %
43 % THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
44 % ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
45 % WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
46 % DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
47 % ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
48 % (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
49 % LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
50 % ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
51 % (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
52 % SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
53 %
54 % Input args to be a vector
55 % vars = {'p', 'Ts', 'Ps', 'fmat'};
56 % for num = 1:length(vars)
57 %     if eval(['~any(size(' vars{num} ')==1)'])
58 %         error(['@drpms/bode: the input ''' vars{num} ...
59 %             ''' must be a vector.']);
60 %     end
61 % end
62 %
63 % if length(fmat)==2&fmat(1)<fmat(2)
64 %     f = fmat(1):fmat(1):fmat(2);
65 % else
66 %     error('@drpms/bode: the style of ''fmat'' has to be [step fmax].');
67 % end
68 %
69 % vars = {'Ts', 'Ps'};
70 % for num = 1:length(vars)

```

```

71         if eval(['length(' vars{num} ') ~= length(p)'])
72             error(['@drpms/bode: the size of ''' vars{num} ...
73                 ''' must be same as the size of '''p'''']);
74         end
75     end
76
77     if ~exist('yscale', 'var')
78         yscale = 'linear';
79     elseif strcmp(yscale, strvcat('linear', 'db', 'log'), 'exact')
80
81     else
82         error(['@drpms/bode: the 5th argument has to be ' ...
83             '''linear'', 'db' or 'log'.'']);
84     end
85
86     for num = 1:length(p)
87         res(:,num) = getresponse(p(num), Ts(num), Ps(num), f);
88         phse(:,num) = phase(res(:,num))/pi*180;
89     end
90     mag = abs(res);
91     %When the outputs are requested, the figure output will be suppressed.
92     no = nargin;
93     if isempty(nargoutchk(0,3,no))
94         if no >= 1
95             return
96         else
97             h = findobj('tag', 'drpms_bode');
98             if isempty(h)
99                 h = figure('name', 'drpms (bode diagram)', ...
100                     'numbertitle', 'off', ...
101                     'tag', 'drpms_bode');
102             end
103             clf(h);
104             figure(h);
105
106             switch yscale
107                 case 'db'
108                     h1(1) = subplot(2,1,1);
109                     plot(f, 20*log10(mag));
110                     ylabel('Amplitude [db]');
111                     h1(2) = subplot(2,1,2);
112                     plot(f, phse);
113                     ylabel('Phase [deg]');
114                     xlabel('Frequency [Hz]');
115                 case {'log', 'linear'}
116                     h1(1) = subplot(2,1,1);
117                     plot(f, mag);
118                     ylabel('Amplitude');
119                     h1(2) = subplot(2,1,2);
120                     plot(f, phse);
121                     ylabel('Phase [deg]');
122                     xlabel('Frequency [Hz]');
123                     set(h1(1), 'YScale', yscale);
124             end
125             set(h1, 'XGrid', 'on', 'YGrid', 'on');
126         end

```

```

127         clear mag phse f
128     end
129
130     function [ phse ] = phase( value )
131
132     phse = atan2(imag(value), real(value));
133     len = length(phse);
134     d = -diff(phse);
135     I = find(abs(d) > pi+0.5);
136
137     if ~isempty(I)
138         for i = I'
139             phse = phse + 2*pi*sign(d(i))*[zeros(i,1); ones(len-i,1)];
140         end
141     end

```

C.6 step.m

```

1  function [y, t] = step(p, Ts, Ps, fmat)
2  % STEP draws or returns the step response of DRPMS object.
3  %
4  % STEP(P, TS, PS, [FRQ_STEP MAX_FRQ]) draws the step response of the DRPMS
5  % object P which represents the pressure measurement system.
6  % P, TS and PS should be vectors with the same length.
7  % TS and PS should be the ambient static temperature the ambient static pressure
8  % of the system.
9  % [FRQ_STEP MAX_FRQ] are transformed to the frequency vector
10 % F = FRQ_STEP:FRQ_STEP:MAX_FRQ
11 % The step response is computed using the informations at the frequency F.
12 %
13 % [Y,T] = STEP(P, TS, PS, [FRQ_STEP MAX_FRQ]) returns the output response Y
14 % and the time vector T. No plot is drawn on the screen. If the length of
15 % P is N, Y is an array of size [1+length(FMAT) N] where Y(:,j) gives the step
16 % response of the j-th object.
17 %
18 % See also drpms, drpms/get, drpms/set, drpms/getresponse, drpms/bode,
19 % drpms/transient, drpms/plus
20 %
21 % Author:
22 % Makoto Ueno (ueno.makoto@jaxa.jp)
23 %
24 % License:
25 %
26 % Copyright (c) 2006, Japan Aerospace Exploration Agency
27 % All rights reserved.
28 %
29 % Redistribution and use in source and binary forms, with or without modification,
30 % are permitted provided that the following conditions are met:
31 % 1. Redistributions of source code must retain the above copyright notice,
32 %    this list of conditions and the following disclaimer.
33 % 2. Redistributions in binary form must reproduce the above copyright notice,
34 %    this list of conditions and the following disclaimer in the documentation
35 %    and/or other materials provided with the distribution.
36 % 3. Neither the name of the Japan Aerospace Exploration Agency nor the names
37 %    of its contributors may be used to endorse or promote products derived from

```

```

38 %      this software without specific prior written permission.
39 %
40 %      THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
41 %      ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
42 %      WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
43 %      DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
44 %      ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
45 %      (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
46 %      LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
47 %      ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
48 %      (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
49 %      SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
50
51 % Input args must be a vector
52 vars = {'p', 'Ts', 'Ps', 'fmat'};
53 for num = 1:length(vars)
54     if eval(['~any(size(' vars{num} ')==1)'])
55         error(['@drpms/step: the input ''' vars{num} ...
56                ''' must be a vector.']);
57     end
58 end
59
60 if length(fmat)==2&fmat(1)<fmat(2)
61     f = fmat(1):fmat(1):fmat(2);
62     flen = length(f);
63     t = (0:ceil(flen/2))/(fmat(2)*2);
64     x0len = ceil(flen/2);
65     tmax = x0len/(fmat(2)*2);
66     x0 = ones(x0len, 1);
67 else
68     error('@drpms/step: the style of ''fmat'' have to be [FRQ_STEP MAX_FRQ].');
69 end
70
71 vars = {'Ts', 'Ps'};
72 for num = 1:length(vars)
73     if eval(['length(' vars{num} ') ~= length(p)'])
74         error(['@drpms/step: the length of ''' vars{num} ...
75                ''' must be same as the size of ''p''.']);
76     end
77 end
78
79 [stepRes, t] = transient(p, Ts, Ps, x0, tmax);
80 no = nargout;
81 if isempty(nargoutchk(0,2,no));
82     if no >= 1
83         y = stepRes;
84         return
85     else
86         h = findobj('tag', 'drpms_step');
87         if isempty(h)
88             h = figure('name', 'drpms (step response)', ...
89                        'numbertitle', 'off', ...
90                        'tag', 'drpms_step');
91         end
92         clf(h);
93         figure(h);

```



```

94
95         plot(t, stepRes);
96         ylabel('Output');
97         xlabel('time [sec]');
98         set(gca, 'XGrid', 'on', 'YGrid', 'on')
99     end
100 end

```

C.7 transient.m

```

1  function [y, t] = transient(p, Ts, Ps, x0, tmax)
2  % TRANSIENT draws or returns the arbitrary input response of DRPMS object.
3  %
4  % TRANSIENT(P, TS, PS, X0, TMAX) draws the arbitrary input response
5  % of the DRPMS object P which represents the pressure measurement system.
6  % P, TS and PS should be vectors with the same length.
7  % TS and PS should be the ambient static temperature the ambient static pressure
8  % of the system. X0 is the actual arbitrary input vector which starts from 0 and
9  % end with the value of 1. TMAX is the endurance time of the input vector.
10 %
11 % [Y,T] = TRANSIENT(P, TS, PS, X0, TMAX) returns the output response Y
12 % and the time vector T. No plot is drawn on the screen. If the length of
13 % P is N, Y is an array of size [1+length(FMAT) N] where Y(:,j) gives the output
14 % response of the j-th object.
15
16 % See also drpms, drpms/get, drpms/set, drpms/getresponse, drpms/bode,
17 % drpms/step, drpms/plus
18
19 % Author:
20 % Makoto Ueno (ueno.makoto@jaxa.jp)
21 %
22 % License:
23 %
24 % Copyright (c) 2006, Japan Aerospace Exploration Agency
25 % All rights reserved.
26 %
27 % Redistribution and use in source and binary forms, with or without modification,
28 % are permitted provided that the following conditions are met:
29 % 1. Redistributions of source code must retain the above copyright notice,
30 % this list of conditions and the following disclaimer.
31 % 2. Redistributions in binary form must reproduce the above copyright notice,
32 % this list of conditions and the following disclaimer in the documentation
33 % and/or other materials provided with the distribution.
34 % 3. Neither the name of the Japan Aerospace Exploration Agency nor the names
35 % of its contributors may be used to endorse or promote products derived from
36 % this software without specific prior written permission.
37 %
38 % THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
39 % ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
40 % WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
41 % DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
42 % ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
43 % (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
44 % LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
45 % ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

```

```

46 % (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
47 % SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
48
49 % Input args must be a vector
50 vars = {'p', 'Ts', 'Ps', 'x0'};
51 for num = 1:length(vars)
52     if eval(['~any(size(' vars{num} ')==1)'])
53         error(['@drpms/transient: the input ''' vars{num} ...
54             ''' must be a vector.']);
55     end
56 end
57 if length(tmax)~=1
58     error(['@drpms/transient: the input arg ''' tmax ''' must be a scalar.']);
59 end
60
61 N = length(x0)-1;
62 fmat(1) = 1/(tmax*4);
63 fmat(2) = (N/tmax)/2;
64 f = fmat(1):fmat(1):fmat(2);
65
66 vars = {'Ts', 'Ps'};
67 for num = 1:length(vars)
68     if eval(['length(' vars{num} ') ~= length(p)'])
69         error(['@drpms/transient: the length of ''' vars{num} ...
70             ''' must be same as the size of ''' p'''.']);
71     end
72 end
73
74 if size(x0)*[0; 1]~=1
75     x0 = x0';
76 end
77 transientY = [zeros(N*2, 1); x0; ones(N, 1)];
78 fTransientY = fft(transientY);
79 for num = 1:length(p)
80     res = getresponse(p(num), Ts(num), Ps(num), f);
81     fTransientRes(:,num) = fTransientY.*[1; res; flipud(conj(res))];
82 end
83 transientRes = ifft(fTransientRes);
84 transientRes = transientRes((length(f)+1):(length(transientRes)-ceil(length(f)/2)),:);
85
86 t = (0:floor(length(f)/2))/(f(end)*2);
87
88 no = nargout;
89 if isempty(nargoutchk(0,2,no));
90     if no >= 1
91         y = transientRes;
92         return
93     else
94         h = findobj('tag', 'drpms_transient');
95         if isempty(h)
96             h = figure('name', 'drpms (transient input response)', ...
97                 'numbertitle', 'off', ...
98                 'tag', 'drpms_transient');
99         end
100         clf(h);
101         figure(h);

```

```

102         subplot(2,1,1)
103         plot(t, x0);
104         ylabel('Input');
105         set(gca, 'XGrid', 'on', 'YGrid', 'on')
106         subplot(2,1,2)
107         plot(t, transientRes);
108         ylabel('Output');
109         xlabel('time [sec]');
110         set(gca, 'XGrid', 'on', 'YGrid', 'on')
111     end
112 end

```

C.8 plus.m

```

1  function p = plus(p1, p2)
2  %
3  %   PLUS Addition of two DRPMS object
4  %
5  %   P = PLUS(P1, P2) performs P = P1 + P2.
6  %   Adding DRPMS objects is equivalent to connecting adjacent
7  %   tube & transducer systems. Both P1 and P2 length must be 1.
8  %   The result depends on the argument order. P1 is on the input side and
9  %   P2 is on the inner side.
10 %
11 %   See also drpms, drpms/get, drpms/set, drpms/getresponse,
12 %   drpms/bode, drpms/step, drpms/transient
13 %
14 %   Author:
15 %   Makoto Ueno (ueno.makoto@jaxa.jp)
16 %
17 %   License:
18 %
19 %   Copyright (c) 2006, Japan Aerospace Exploration Agency
20 %   All rights reserved.
21 %
22 %   Redistribution and use in source and binary forms, with or without modification,
23 %   are permitted provided that the following conditions are met:
24 %   1. Redistributions of source code must retain the above copyright notice,
25 %      this list of conditions and the following disclaimer.
26 %   2. Redistributions in binary form must reproduce the above copyright notice,
27 %      this list of conditions and the following disclaimer in the documentation
28 %      and/or other materials provided with the distribution.
29 %   3. Neither the name of the Japan Aerospace Exploration Agency nor the names
30 %      of its contributors may be used to endorse or promote products derived from
31 %      this software without specific prior written permission.
32 %
33 %   THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
34 %   ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
35 %   WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
36 %   DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
37 %   ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
38 %   (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
39 %   LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
40 %   ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
41 %   (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS

```

```

42 % SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
43
44     if length(p1)~=1 | length(p2)~=1
45         error('Both length of P1 and P2 have to be 1.');
```

```

46     else
47         num = 1;
48         q(num) = p1;
49         while ~isempty(get(q(num), 'nextSystem'))
50             num = num + 1;
51             q(num) = get(q(num-1), 'nextSystem');
```

```

52         end
53         p = set(q(num), 'nextSystem', p2);
54         for num2 = (num-1):-1:1
55             p = set(q(num2), 'nextSystem', p);
56         end
57     end
58

```

C.9 private/viscosity.m

```

1  function mu = viscosity(T, T0, mu0, S)
2  % VISCOSITY returns the viscosity of air.
3  %
4  % MU = VISCOSITY(T, T0, mu0, S) returns the viscosity of arbitrary gas based
5  % on the Sutherland's law.
6  % T is absolute temperature, T0 and mu0 are the temperature and the viscosity
7  % at which the Sutherland's constant for the gas is defined. S is the
8  % Sutherland's constant.
9  %
10 % This is a private helper function of the class 'drpms'.
11 %
12 % See also DRPMS
13
14 % Author:
15 % Makoto Ueno (ueno.makoto@jaxa.jp)
16 %
17 % License:
18 %
19 % Copyright (c) 2006, Japan Aerospace Exploration Agency
20 % All rights reserved.
21 %
22 % Redistribution and use in source and binary forms, with or without modification,
23 % are permitted provided that the following conditions are met:
24 % 1. Redistributions of source code must retain the above copyright notice,
25 % this list of conditions and the following disclaimer.
26 % 2. Redistributions in binary form must reproduce the above copyright notice,
27 % this list of conditions and the following disclaimer in the documentation
28 % and/or other materials provided with the distribution.
29 % 3. Neither the name of the Japan Aerospace Exploration Agency nor the names
30 % of its contributors may be used to endorse or promote products derived from
31 % this software without specific prior written permission.
32 %
33 % THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
34 % ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
35 % WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

```

```

36 %   DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
37 %   ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
38 %   (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
39 %   LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
40 %   ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
41 %   (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
42 %   SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
43
44 mu = mu0 * (T/T0).^(3/2) .* ((T0+S)./(T+S));

```

C.10 private/airconductivity.m

```

1  function TLambda = airconductivity()
2  % AIRCONDUCTIVITY returns the thermal conductivity table of air.
3  %
4  %   TLAMBDA = AIRCONDUCTIVITY()
5  %
6  %   This is a private helper function of the class 'drpms'.
7  %   No input args are required.
8  %   The function returns the table of the air conductivity with respect to absolute
9  %   temperature.
10 %
11 %   The air conductivity is based on Kadoya, K., Matsunaga, N. and Nagashima, A.,
12 %   "Viscosity and Thermal Conductivity of Dry Air in the Gaseous Phase,"
13 %   J. Phys. Chem. Ref. Data ,14, 4 (1985) 947.
14 %
15 %   See also DRPMS
16
17 %   Author:
18 %   Makoto Ueno (ueno.makoto@jaxa.jp)
19 %
20 %   License:
21 %
22 %   Copyright (c) 2006, Japan Aerospace Exploration Agency
23 %   All rights reserved.
24 %
25 %   Redistribution and use in source and binary forms, with or without modification,
26 %   are permitted provided that the following conditions are met:
27 %   1. Redistributions of source code must retain the above copyright notice,
28 %   this list of conditions and the following disclaimer.
29 %   2. Redistributions in binary form must reproduce the above copyright notice,
30 %   this list of conditions and the following disclaimer in the documentation
31 %   and/or other materials provided with the distribution.
32 %   3. Neither the name of the Japan Aerospace Exploration Agency nor the names
33 %   of its contributors may be used to endorse or promote products derived from
34 %   this software without specific prior written permission.
35 %
36 %   THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
37 %   ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
38 %   WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
39 %   DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
40 %   ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
41 %   (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
42 %   LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
43 %   ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

```

```
44 % (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
45 % SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
46
47     T = [150; 200; 250; 273.15; 300; 320; 360; 400; 500; ...
48           600; 800; 1000; 1200; 1500; 2000];
49     lambda = [14.07; 18.36; 22.41; 24.21; 26.23; 27.70; 30.54; 33.28; 39.71; ...
50              45.73; 56.99; 67.63; 77.92; 92.96; 117.5]*1e-3;
51     TLambda = [T lambda];
52
```

参考文献

- [1] Bergh, H. and Tijdeman, H., “Theoretical and Experimental Results for the Dynamic Reponse of Pressure Measuring Systems,” NLR-TR F.238, Jan. 1965.
- [2] Tijdeman, H., “Investigations of the Transonic Flow Around Oscillating Airfoils,” NLR-TR 77090 U, 1977.
- [3] Chapin, William G., “Dynamic-Pressure Measurements Using an Electronically Scanned Pressure Module,” NASA TM-84650, July. 1983.
- [4] Tijdeman, H. and Bergh, H., “The Influence of the Main Flow on the Transfer Function of Tube-transducer Systems Used for Unsteady Pressure Measurements,” NLR-MP 72023 U, 1972.
- [5] The Mathworks Inc., “Getting Started with MATLAB,” 2004.
- [6] Kadoya, K., Matsunaga, N. and Nagashima, A., “Viscosity and Thermal Conductivity of Dry Air in the Gaseous Phase,” J. Phys. Chem. Ref. Data ,14, 4 (1985) 947.

宇宙航空研究開発機構研究開発報告 JAXA-RR-05-037

発行 平成18年3月31日
編集・発行 宇宙航空研究開発機構
〒182-8522 東京都調布市深大寺東町7-44-1
URL : <http://www.jaxa.jp/>
印刷・製本 (株)ビー・シー・シー・

本書及び内容についてのお問い合わせは、下記にお願いいたします。

宇宙航空研究開発機構 情報システム部 研究開発情報センター
〒305-8505 茨城県つくば市千現2-1-1
TEL:029-868-2079 FAX:029-868-2956

© 2006 宇宙航空研究開発機構

※ 本書の一部または全部を無断複写・転載・電子媒体に加工することを禁じます。

