平成20年度
長岡技術科学大学
工学部 課題研究論文

ロバスト制御のための遠隔実験環境の開発と能動騒音制御への応用

Remote Experiment Environment for Robust Control System with an Application to Active Noise Control

長岡技術科学大学　工学部
機械創造工学課程　07303483

著者氏名　北田勇

指導教員　小林泰秀

平成21年2月24日

要旨

近年、システム制御教育における受講者の理解度を向上させるために、遠隔実験環境（リモートラボラトリ）が提案されている。既存システムではアームロボットの PID 制御等、比較的簡単な制御問題が扱われており、扱うパラメータも少ないが、現実的にはより複雑な制御対象を扱うことが多い。そこで本論文では、パラメータの多い、より自由度の高い制御問題を扱った、ロバスト制御における遠隔実験環境を開発することを目的としている。本論文では、その題材としてダクトの能動騒音制御を扱い、以下の 2 点について取り組んだ。

1. CUI ベースの教材と GUI ベースの教材の比較
2. GUI ベースの教材の改善

1. に関して

本研究室では、実験装置を制御するサーバをコマンドラインで直接操作する CUI ベースの実験教材と、ブラウザからサーバにアクセスしマウスクリックを中心に作業を行う GUI ベースの教材の 2 つのシステムが開発されている。本論文では、学生実験により 2 つのシステムの比較を行った。その結果、GUI ベースの教材は CUI ベースの教材に比べ以下の点で優れていることが明らかになった。

- キーボード入力が無くなるため、実験工程において作業時間が短縮される
- ブラウザのみを用いるため、実験に直接関係のないツールによる負担が無くなる

2. に関して

今後システムを遠隔化する際に、臨場感を得るための工夫として、web カメラを設置し、実験装置が動作しているか確認できるようにした。

また音声 dat ファイルの wav 化を行い、システム内に組み込むことにより、消音実験結果を web 上で再生、ダウンロード可能にした。

目次

1 諸言 **11**

2 １次元ダクトの騒音制御実験 **13**
2.1 システムの構成 13
2.2 実験手順 14
2.2.1 実験の流れ 14
2.2.2 CUI での作業内容 16
2.2.3 GUI での作業内容 18

3 システムの評価 **27**
3.1 評価方法 27
3.2 結果 28
3.2.1 実験を行う工程 40
3.2.2 図の表示のみを行う工程 40
3.2.3 数値入力を行う工程 41
3.2.4 会話による比較 43
3.2.5 まとめ 46

4 GUI の改善 **47**
4.1 現状システムの改善点 47
4.2 web カメラの設置 48
4.3 音声 dat ファイルの wav 化 49

5 結言 **51**

6 今後の展望 **53**

A 元データ（解析シート） **59**

B 解析メモ **61**

C 元データ（会話の記録） **63**

図一覧

2.1 ダクト外観 13
2.2 モデル 14
2.3 現状のシステム 14
2.4 Gyu(s) のモデル化誤差と重み関数のゲイン特性の例 15
2.5 スケーリング d と変数 k の例 15
2.6 CUI の PC 画面 16
2.7 GUI の PC 画面 18
2.8 周波数応答実験 (GUI) 19
2.9 ノミナルプラント作成 GUI) 20
2.10 ノミナルプラントー図の確認 (GUI) 20
2.11 重み関数の設定 (GUI) 21
2.12 重み関数の確認 (GUI) 21
2.13 コントローラの設計 (GUI) 22
2.14 コントローラの設計ー図の確認, 性能予測 (GUI) 22
2.15 性能予測, 消音制御実験 (GUI) 23
2.16 消音制御系の性能評価 (GUI) 24
2.17 性能予測ー図の確認, データの保存 (GUI) 24
2.18 重み関数の設定へ戻る (GUI) 25

3.1 解析シート 28
3.2 集計結果 (CUI1) 29
3.3 集計結果 (CUI2) 30
3.4 集計結果 (GUI1) 31
3.5 集計結果 (GUI2) 32
3.6 集計結果 (GUI3) 33
3.7 作業時間の比較 39
3.8 作業時間内訳（ノミナルプラント作成） 40
3.9 作業時間内訳（消音制御系の性能予測） 41
3.10 作業時間内訳（消音制御系の性能評価） 41
3.11 重み関数設定における XEmacs の割合 42
3.12 コントローラ設計における XEmacs の割合 42
3.13 会話内容の割合 46

4.1 目標となるシステム例 47

4.2 ブラウザの画面（周波数応答実験） 48
4.3 ブラウザの画面（結果閲覧のページ） 49

表一覧

3.1 集計まとめ (CUI1) 34
3.2 集計まとめ (CUI2) 35
3.3 集計まとめ (GUI1) 36
3.4 集計まとめ (GUI2) 37
3.5 集計まとめ (GUI3) 38
3.6 会話集計 (CUI1) 43
3.7 会話集計 (CUI2) 44
3.8 会話集計 (GUI1) 44
3.9 会話集計 (GUI2) 45
3.10 会話集計 (GUI3) 45

第 1 章

諸言

工学教育においては、講義などの座学とともに、実験や実習などを通し体験学習をすることは重要である。近年、システム制御教育における受講者の理解度を向上させるために、遠隔実験環境（リモートラボラトリ）が提案されている。遠隔実験環境とは、実験現場から離れた、遠隔地から実験を行える環境のことを言う。遠隔実験環境を利用すれば、実験ができない環境（機材の不足、コスト等）からでも実験が行えるという利点がある。近年はインターネットの普及も進んでいるため、遠隔実験環境はより実用的になっていると言えよう。

既存システムではアームロボットの PID 制御 [1][2]、モータの角度制御 [3] 等、比較的簡単な制御問題が扱われており、扱うパラメータも少ないが、現実的にはより複雑な制御対象を扱うことが多い。そこで本論文では、パラメータが多く、より自由度の高い制御問題を扱える、ロバスト制御における遠隔実験環境を開発することを目的としている。本論文ではその題材としてダクトの能動騒音制御を扱った。

提案するシステムでは、上述のように複雑な制御対象であるほか、音を扱っているためモータやアームロボットなどと比べ、実験現場での機械的なトラブルは少なく、遠隔化のリスクが小さいという利点がある。

本論文では目的となる遠隔実験環境開発のために、以下の 2 点について取り組んだ。

- CUI ベースの教材と GUI ベースの教材の比較
- GUI ベース教材の改善

本論文の構成は次の通りである。第 2 章で対象となるシステム、実験の手順を示す。このシステムを第 3 章で比較、検討し、第 4 章で改善した点を述べる。第 5 章で結言を述べ、第 6 章では今後の展望を述べる。

第 2 章

１次元ダクトの騒音制御実験

制御対象となるダクトの騒音制御実験システムについて述べる。

2.1　システムの構成

図 2.1 にダクトの外観、図 2.2 に実験装置のモデルを示す。ダクトは直径約 10cm、長さ約 2m の塩ビ管製で、右端にはファンの騒音を模擬するためのスピーカ (SPK1) が取り付けられている．SPK1 は、パソコン (PC) で生成された外乱信号 w により、ローパスフィルタ (LPF) とパワーアンプ (AMP) を介して駆動される。騒音は、リファレンスマイクにより検出され、プリアンプ (AMP) とローパスフィルタを通して観測出力 y としてパソコンに入力される。この情報に基づき，適切な制御入力 u を計算し、制御音発生用のスピーカ (SPK2) を駆動する。騒音と制御音は破壊的干渉 (destructive interference) を起こし、SPK2 より下流側 (図では左側) では、管内の音圧レベルが低減される。その結果はエラーマイクを介して、評価出力 z としてパソコンに入力される。

Fig. 2.1: ダクト外観

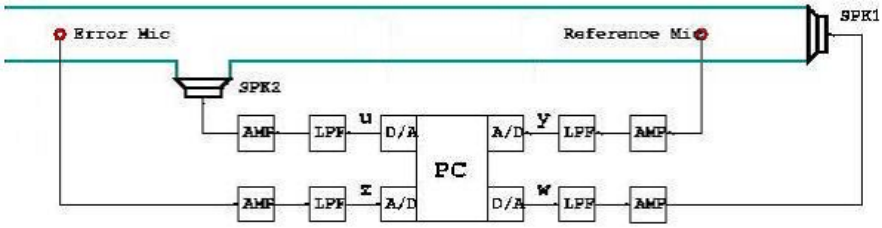


Fig. 2.2: モデル

実験制御用 PC では OS に Linux を使用しているほか、各種演算に Matlab を用いている。本研究室では、これらを説明を見ながらコマンドラインで直接操作する CUI(Character User Interface) ベースの教材（以下 CUI）と、ブラウザのみで動作しマウスクリックを中心に作業を進めていく GUI(Graphical User Interface) ベースの教材（以下 GUI）という 2 つのシステムが開発されている (図 2.3)。

(a) *CUI* (b) *GUI*

Fig. 2.3: 現状のシステム

2.2 実験手順

実験の手順について述べる。はじめに CUI,GUI 共通である実験の流れを述べ、次にそれぞれの具体的な作業内容について述べる。

2.2.1 実験の流れ

1. 周波数応答実験
   10kHz から 2kHz まで、周波数を変化させながら周波数応答データを取得させ、その結果がデータファイル spk1.dat, spk2.dat に格納される。spk1.dat には、一次音源 (w) から エラーマイク (z)、リファレンスマイク (y) までの周波数応答を記録する。spk2.dat には、二次音源 (u) から エラーマイク (z)、リファレンスマイク (y) までの周波数応答を記録する。

2. ノミナルプラントの作成
   周波数応答データ (spk1.dat, spk2.dat) を近似して、ノミナルプラントの伝達関数を求める。周波数応答実験の結果を近似して、4 つの伝達関数 Gzw(s), Gyw(s), Gzu(s), Gyu(s) を求め、その Bode 線図を周波数応答実験結果に重ねて表示する。実験点と近似曲線が大体重なっていることを確認する。

3. 重み関数の設定
   モデル化誤差を表すために加法的摂動モデルを用いて重み関数（モデル化誤差のゲイン特性を表す伝達関数）を設定する。制御系が安定となるために、全周波数帯域で、モデル化誤差のゲイン特

性よりも重み関数のゲイン特性が大きくなるように 重み関数を決定する。決定した重み関数に基づき、Gyu(s) のモデル化誤差と重み関数のゲイン特性が表示されるので (図 2.4)、これを見ながら、適切な結果が得られるまで重み関数の設定を繰り返す。

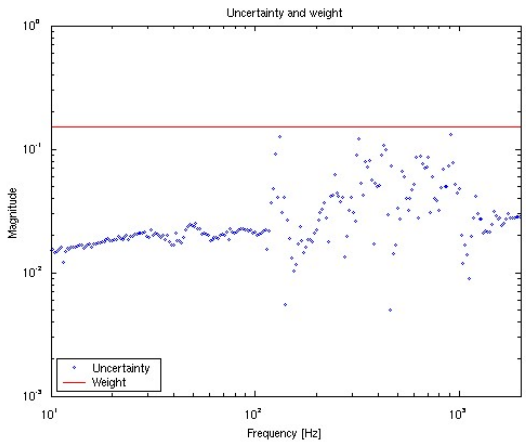


Fig. 2.4: Gyu(s) のモデル化誤差と重み関数のゲイン特性の例

4. コントローラの設計
定数スケールド H ∞制御問題を解き、望ましい補償器を設計する。スケーリングを変化する範囲、刻み幅を指定し、補償器が設計されると、スケーリング d と変数 k のグラフ (図 2.5)、補償器の Bode 線図、閉ループ系の最大特位値のグラフが 表示される。スケーリングを変化する範囲が適切であれば、d と k のグラフはおおよそ上に凸の形になる。d と k のグラフを基に、適切な補償器が得られるまで作業を繰り返す。なお、刻み幅は細かく すればするほど精度の高い補償器が得られるが、設計に時間がかかる。
補償器の Bode 線図、閉ループ系の最大特位値のグラフは後に行われる消音制御に関連し、考察の材料となる。

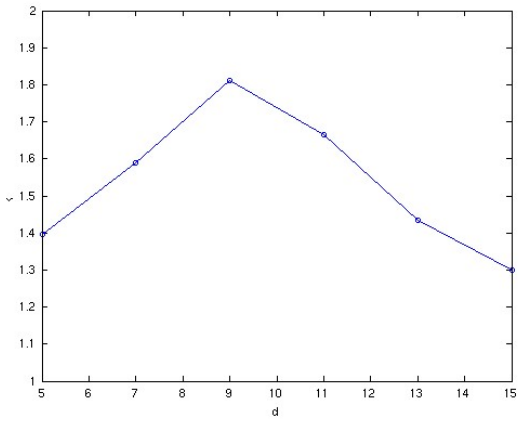


Fig. 2.5: スケーリング d と変数 k の例

5. 消音制御系の性能予測
一巡伝達関数のナイキスト軌跡、w から z までのゲイン特性の 2 つの図が得られる。この図から、ナイキストの安定判別により閉ループ系の安定性を確認する。また Gzw(s) のゲイン特性と、補償器込みの閉ループ系のゲイン特性を比較することにより、消音制御系の性能を予測する。

6. 消音制御実験
得られた補償器を用いて、消音制御実験を行う。実験は 10 秒間で、前半 5 秒が制御なし、後半 5 秒が制御ありで行われる。実験データは result.dat に格納される。

7. 消音制御系の性能評価
消音制御の結果 (result.dat) から、時間応答と周波数分析 (FFT) 結果を 表示する。時間応答グラフでは z,y,u,w の 4 つのグラフが表示され、制御ありとなしで何が違うか考察を行う。FFT 分析結果では、z,w,z/w の 3 つのグラフが表示され、消音制御性能の予測結果と比較、検討を行う。

8. 実験データの保存
重み関数を変更し再度実験を行う場合、予め、変更前に行った実験結果が上書きされて消えないように、データディレクトリをコピーする

重み関数の設定に戻る。

2.2.2 CUI での作業内容

CUI では全工程にわたってコマンドで作業を進めていく。その際、使うことになるツールは、各種実験、ディレクトリの作成や管理を行うターミナルである kterm、コントローラ設計などの演算を行う Matlab、ファイルの中身を編集するエディタの XEmacs の 3 つである。web 上に実験の進め方（コマンド内容や何を行っているかなど）が書かれたページがあるので、それを見ながら実験を進めていく。

なお実際の画面は図 2.6 のように、ブラウザ、XEmacs、kterm × 2 となっているが、片方の kterm は Matlab を動かしている。混同を避けるため、以下 Matlab を動かしている kterm は Matlab と呼ぶ。

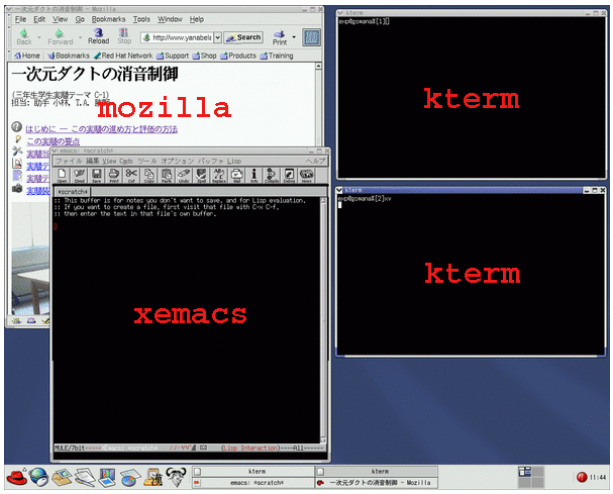


Fig. 2.6: CUI の PC 画面

実際に行う作業は次の通りである。kterm では%以下、Matlab では >> 以下、XEmacs では下線部が実際に入力するコマンド（ファイル名）である。

1. 周波数応答実験 kterm 上で作業を行う。
   (a) ディレクトリ移動 % cd ~/WWW/1
   (b) 周波数応答実験用モジュールの組み込む % insmod freqresp_module.o
   (c) モジュールが組み込まれたことを確認 % lsmod
   (d) 周波数応答実験開始 % ./freqresp_app
   (e) モジュールを取り除く % rmmod freqresp_module
   (f) モジュールが取り除かれたことを確認 % lsmod
2. ノミナルプラントの作成 Matlab 上で作業を行う。
   (a) nominal.m を実行 >> nominal
3. 重み関数の設定
   (a) weight.m を実行し重み関数を設定 >> weight
   表示される図（2.4）を参考に、weight.m を編集する。XEmacs で以下の作業を行う。
   (b) ファイルを開く。ファイル名を入力 Find file: /WWW/1/weight.m
   (c) ファイルを編集 w = 0.5(例)
   (d) ファイルを上書き保存
   (e) 適切な結果が得られるまで繰り返す
4. コントローラの設計 Matlab 上で作業を行う。
   (a) hinf.m を実行し、コントローラを設計 >> hinf
   表示される図（2.5）を参考に、hinf.m を編集する。XEmacs で以下の作業を行う。
   (b) ファイルを開く。ファイル名を入力 Find file: /WWW/1/hinf.m
   (c) ファイルを編集 for d = 5:2:15(例)
   (d) ファイルを上書き保存
   (e) 適切な結果が得られるまで繰り返す
5. 消音制御系の性能予測 Matlab 上で作業を行う。
   (a) ナイキストの安定判別、Gzw(s) と補償器込みの閉ループ系とのゲイン特性のグラフを表示
   >> conmpare_perf
6. 消音制御実験 kterm 上で作業を行う。
   (a) 消音制御実験用モジュールを組み込む % insmod hinf_module.o
   (b) モジュールが組み込まれたことを確認 % lsmod
   (c) 消音制御実験開始 % ./hinf_app
   (d) 消音制御実験用モジュールを取り除く % rmmod hinf_module
   (e) モジュールが取り除かれたことを確認 %lsmod

7. 消音制御系の性能評価 Matlab 上で作業を行う。
   (a) 実験データを読み込む >> load result.dat;
   (b) 消音制御結果を時間応答のグラフに表示 >> perf_time
   (c) 周波数分析のグラフを表示 >> perf_fft
8. 実験データの保存
   kterm 上で作業を行う。
   (a) データディレクトリのコピー % cp -a data data1

2.2.3 GUI での作業内容

次に GUI での作業内容について述べる。GUI 教材ではブラウザのみを用いて、クリックで実験を進めていく。こちらも、前述の説明ページを参考にしながら実験を行う（図 2.7）。ブラウザの右側が説明ページとなっている。

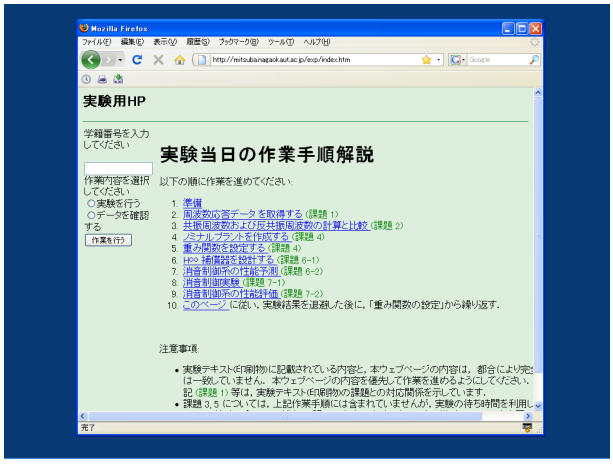


Fig. 2.7: GUI の PC 画面

実際に行う作業は次の通りである。

1. 周波数応答実験

   (a) ボタンをクリックすると周波数応答実験が行われる（図 2.8）

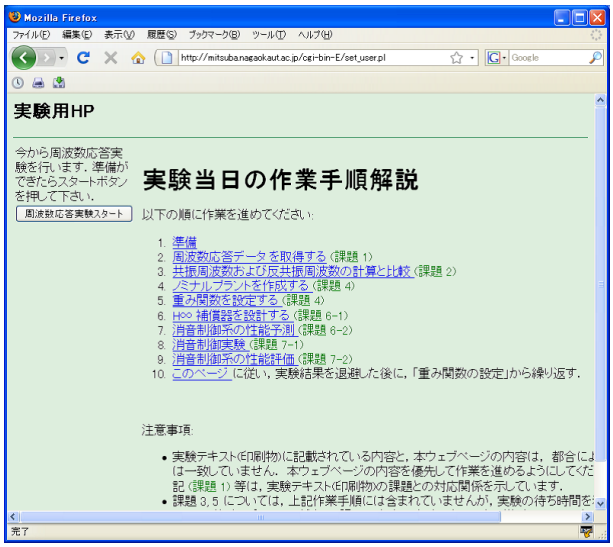


Fig. 2.8: 周波数応答実験 (GUI)

2. ノミナルプラントの作成

   (a) ボタンをクリックするとノミナルプラントが作成される（図 2.9）

   (b) リンクをクリックすると Gzw(s), Gyw(s), Gzu(s), Gyu(s) の Bode 線図が表示される。図をクリックすると拡大。 ボタンクリックで次の工程へ（図 2.10）

3. 重み関数の設定

   (a) 数値を入力し、ボタンをクリックすると重み関数が設定される（図 2.11）

   (b) Gyu(s) のモデル化誤差と重み関数のゲイン特性が表示される。(図 2.4) 図をクリックすると拡大。 上のボタンでやり直し、下のボタンで次の工程へ（図 2.12）

4. コントローラの設計

   (a) 数値を入力し、ボタンをクリックするとコントローラが設計される（図 2.13）

   (b) スケーリング d と変数 k のグラフ (図 2.5)、 補償器の Bode 線図、閉ループ系の最大特位値のグラフが 表示される。図をクリックすると拡大。上のボタンをクリックするとやり直し、下のボタンは次の工程のボタン（図 2.14）

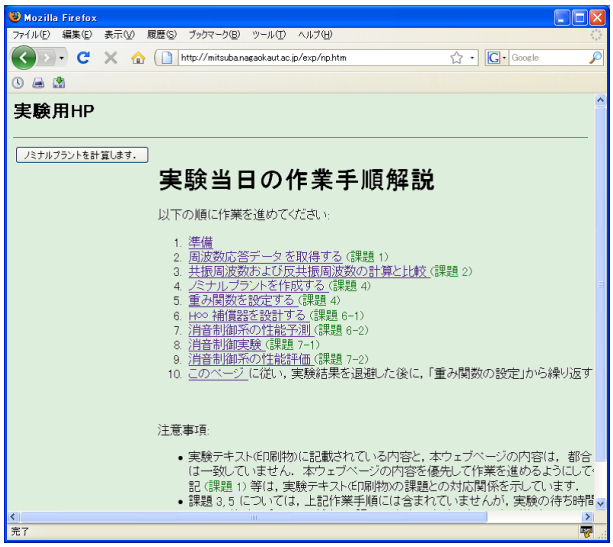


Fig. 2.9: ノミナルプラント作成 GUI)

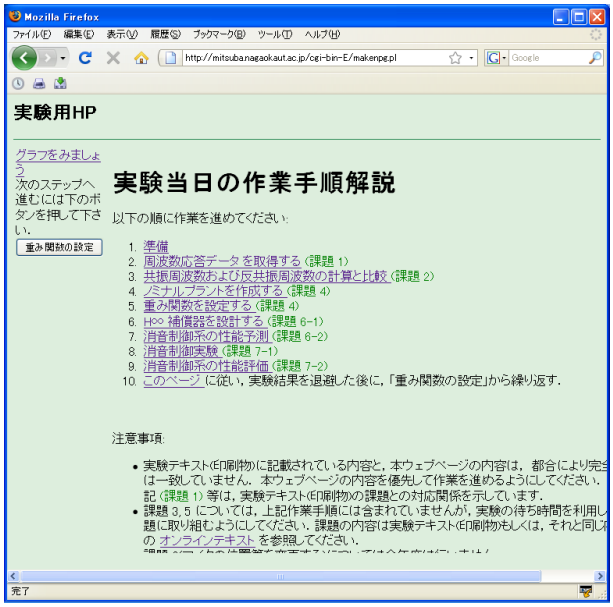


Fig. 2.10: ノミナルプラント一図の確認 (GUI)

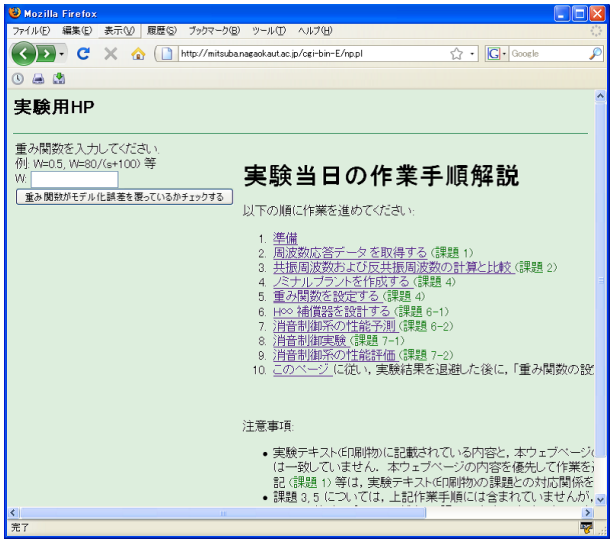


Fig. 2.11: 重み関数の設定 (GUI)

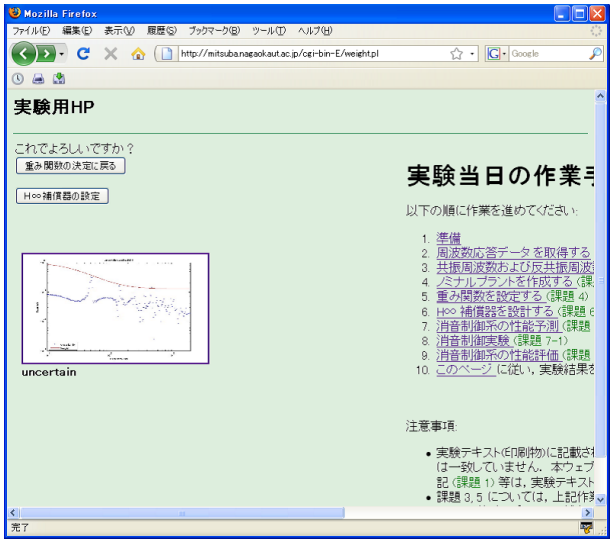


Fig. 2.12: 重み関数の確認 (GUI)

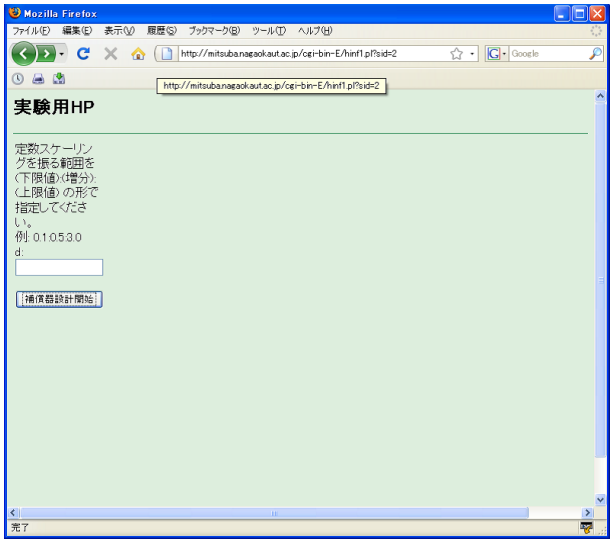


Fig. 2.13: コントローラの設計 (GUI)

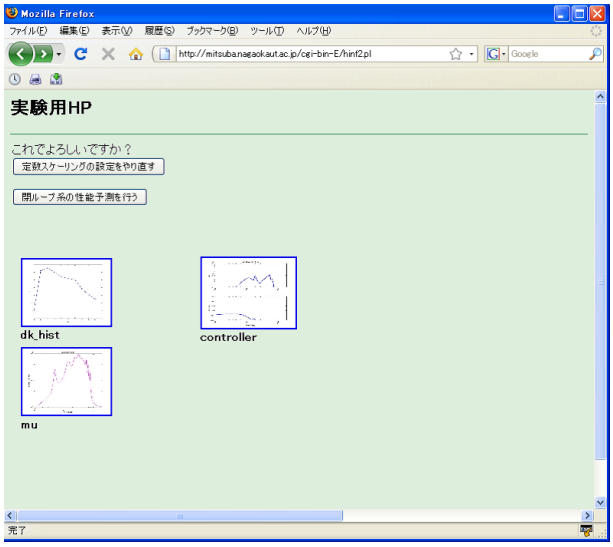


Fig. 2.14: コントローラの設計ー図の確認, 性能予測 (GUI)

5. 消音制御系の性能予測

   (a) 下のボタンをクリックすると、 次のページへ（図 2.14）

   (b) 一巡伝達関数のナイキスト軌跡、w から z までのゲイン特性のグラフが表示される図をクリックすると拡大。ボタンは次の工程のもの（図 2.15）

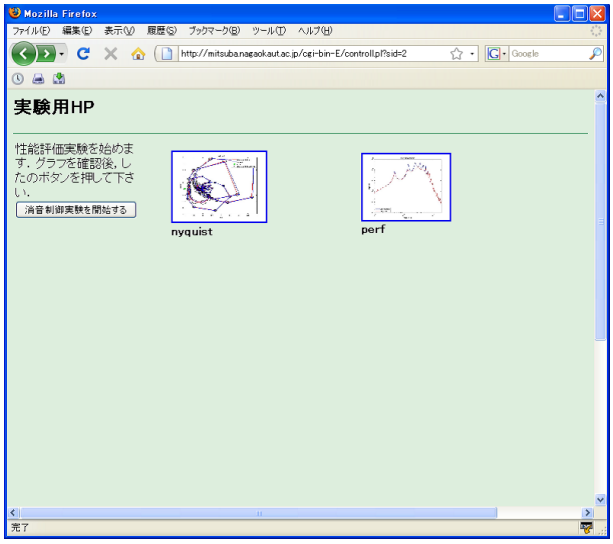


Fig. 2.15: 性能予測, 消音制御実験 (GUI)

6. 消音制御実験

   (a) ボタンをクリックすると、消音制御実験が開始される（図 2.15）
   終了後、自動的に次の工程へ

7. 消音制御系の性能評価

(a) 上のボタンをクリックすると、再び消音制御実験を行う。下のボタンをクリックすると次のページへ（図 2.16）

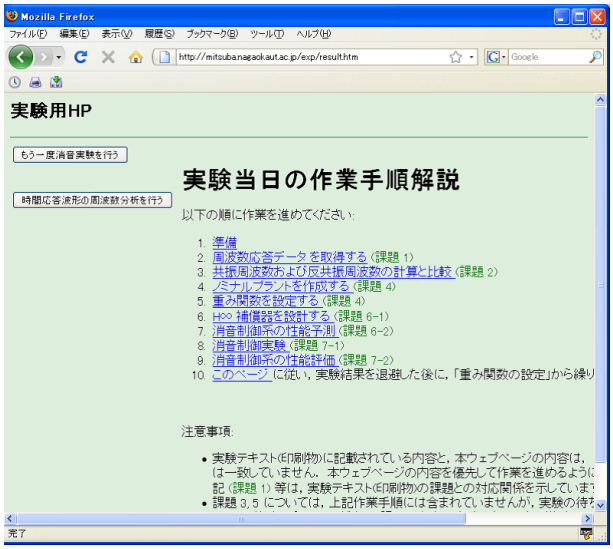


Fig. 2.16: 消音制御系の性能評価 (GUI)

(b) 上のリンクが消音制御結果、下のリンクが周波数分析結果になっており、詳しく見ることができる。ボタンは次の工程のもの（図 2.17）

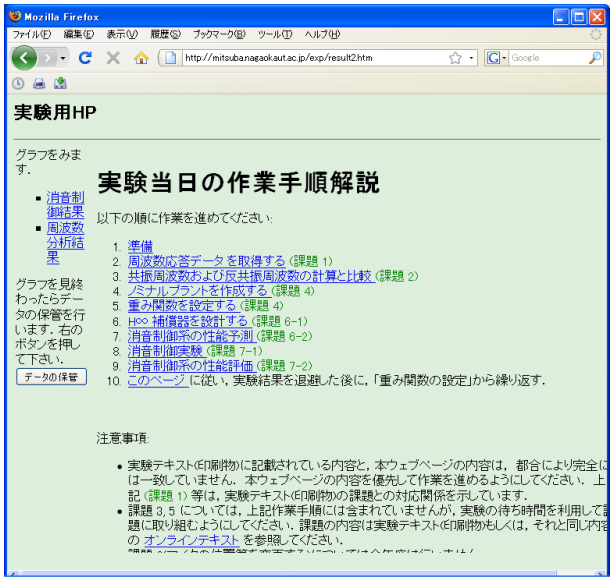


Fig. 2.17: 性能予測ー図の確認, データの保存 (GUI)

8. 実験データの保存

   (a) 下のボタンをクリックすると実験データを保存する。自動的に次のページへ（図 2.17）

   (b) 上のボタンで重み関数の設定へ、下のボタンではじめに戻る（図 2.18）

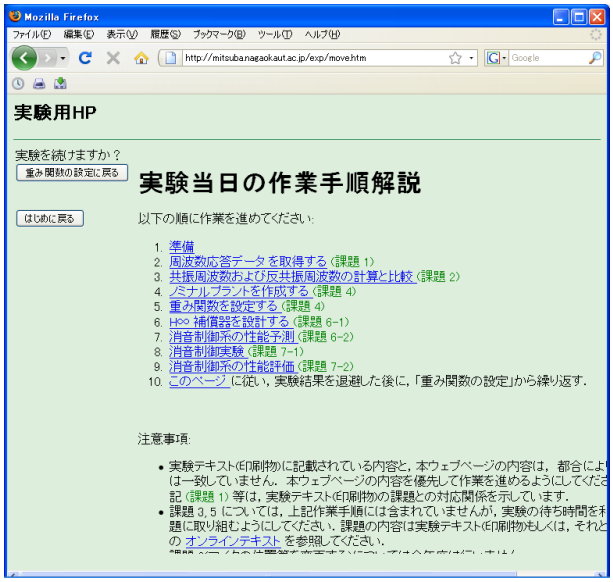


Fig. 2.18: 重み関数の設定へ戻る (GUI)

第 3 章

システムの評価

現状の２つのシステムを用いて、学生実験を行い比較を行った。

3.1　評価方法

学生実験の様子をビデオ撮影により記録、解析を行った。学生実験は平成 20 年度本学学部 3 年生に実施した「機械創造工学実験及び考究」である。5/9(金)，5/20(火)，6/3(火)，6/17(火)，7/1(火) の 5 日間，午後 3〜5 限行われ、前半 2 回が CUI、後半 3 回が GUI を用いものである。以下、日付の若い順に CUI1,CUI2,GUI1,GUI2,GUI3 と呼ぶ。

ビデオはパソコンのモニタと被験者の手元が映るように設置し、画面の変化やキーボード入力の有無を毎秒記録した。記録は専用の解析シートを作成し、そのシートに記録することで解析を行った。図 3.1 は作成したシートの一部である。シートの横軸は時間になっており、0 秒から始まり実験が終了するまで秒単位のマスになっている。撮影した映像から、時刻毎に PC 画面に映っているウィンドウ、ブラウザの画面、図、またキーボード入力の有無を記録し、得られたデータを工程別に分け、それぞれの工程毎の時間を集計する。それらを比較・検証することで評価を行う。なお、解析シートは実験日によって形式が変わっているが、これは記録・集計作業の効率化のためであり、解析方法や結果に影響はない。図 3.1 は最終的なものである。

また、学生同士（被験者とそれ以外）の会話も記録した。音声はビデオカメラに録音されていたもので、会話が始まっているおおよその時刻と、発言内容を可能な限り記録した。記録した発言は、CUI のツールの操作に関するもの、実験そのものに対するもの、その他（操作、実験どちらか判別できないものを含む)、不明（聞き取れなかったもの）の 4 つに分類し、その数を集計した。

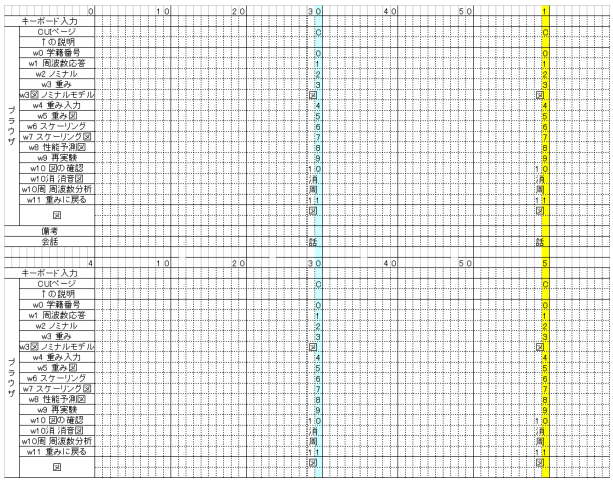


Fig. 3.1: 解析シート

3.2 結果

図 3.2,3.3,3.4,3.5, 3.6 は解析シートに記録したデータを工程別に集計したものである。また表 3.1,3.2,3.3, 3.4,3.5 は集計結果をさらにまとめたものである。元のデータである解析シートの記録、シートの見方のメモを付録に記載する。これらのデータを元に以下の比較・検証を行っていく。ただし、CUI では周波数応答実験を 2 回、GUI では行っていない共振周波数の計算を行っているが、平等な比較を行うため、周波数応答実験時間は 2 回の平均とし、共振周波数の計算に関しては除外する。

図 3.7 はそれぞれの工程別の作業時間を比較したものである。ここで言う作業時間とは、所要時間から待ち時間（コマンド入力後、もしくはクリック後に図が現れたり、画面が変わったりするまでの時間）を引いた時間である。また、横軸の実験回数は、実験を一通り行い、重み関数を設定しなおすところから 2 回目としている

	工程	実際の画面	時間[s]																		
所要時間	1	0,1	246										152			94					
	2	2	1875	1220	655																
	3	3	1087			628		459													
	4	4	147				147														
	5	5	884						688								196				
	6	6	1841							1647								194			
	7	7	212								149								63		
	8	8	533									253		162						118	
	9	9	580												376						204
作業時間	1	0,1	246										152			94					
	2	2	818	580	238																
	3	3	1070			617		453													
	4	4	136				136														
	5	5	859						673								186				
	6	6	1714							1560								154			
	7	7	207								146								61		
	8	8	523									253		152						118	
	9	9	570												370						200
入力時間	1	0,1	26										0			26					
	2	2	220	151	69																
	3	3	152			125		27													
	4	4	5				5														
	5	5	62						40								22				
	6	6	184							159								25			
	7	7	30								16								14		
	8	8	124									28		59						37	
	9	9	60												32						28
入力回数	1	0,1	0																		
	2	2	0																		
	3	3	0																		
	4	4	0																		
	5	5	0																		
	6	6	0																		
	7	7	0																		
	8	8	0																		
	9	9	0																		
1サイクル	1		5846																		
	2		784																		
	3		775																		
	計		7405																		
待ち時間	1	0,1	0										0			0					
	2	2	1057	640	417																
	3	3	17			11		6													
	4	4	11				11														
	5	5	25						15								10				
	6	6	127							87								40			
	7	7	5								3								2		
	8	8	10									0		10						0	
	9	9	10												6						4
スケーリング				64'20	114'20		63		28												
試行錯誤回数				70'20	115'20		365		68												
				80'50	116'10		628		46												
				88'00			432	0入力													
				88'50			45														
Emacs操作時間(重み)				80																	
(重み関数入力以外)				58																	

Fig. 3.2: 集計結果 (CUI1)

	工程	実際の画面	時間[s]																
所要時間	1	0,1	245	100		145													
	2	2	1255		675		580												
	3	3	1035					1035											
	4	4	70						70										
	5	5	1174							703							471		
	6	6	1324								304			654				366	
	7	7	124									93			31				
	8	8	668										270			225			173
	9	9																	
作業時間	1	0,1	245	100		145													
(待ち時間、TAの説	2	2	378		245		133												
明含まない)	3	3	1025					1025											
	4	4	54						54										
	5	5	1089							635							454		
	6	6	993								276			428				289	
	7	7	89									89			0				
	8	8	562										270			119			173
	9	9																	
入力時間	1	0,1	79	47		32													
	2	2	152		91		61												
	3	3	252					252											
	4	4	4						4										
	5	5	72							42							30		
	6	6	200								48			99				53	
	7	7	16									16			0				
	8	8	183										69			54			60
	9	9																	
入力回数	1	0,1	6	4		2													
	2	2	12		7		5												
	3	3	15					15											
	4	4	1						1										
	5	5	20							10							10		
	6	6	25								4			14				7	
	7	7	16									1			?				
	8	8	26										8			8			10
	9	9																	
1サイクル	1		3975(66'15)																
(待ち時間、TAの説	2		910(15'10)																
明含む)	3		1010(16'50)																
	計		5895																
待ち時間	1	0,1	0	0		0													
	2	2	877		430		447												
	3	3	10					10											
	4	4	16						16										
	5	5	85							68							17		
	6	6	242								28			137				77	
	7	7	4									4			?				
	8	8	0										0			?			0
	9	9																	
スケーリング回数		スケーリング回数		58'20		71'40		92'00		40	92	87							
				61'10		72'40		94'00		179	56	115							
						74'30		96'10			108	129							
						75'45		97'20			80	74							
						77'00					81								
						78'00					50								
						79'00					61								
重み		Emacs操作時間		72															
				46															

Fig. 3.3: 集計結果 (CUI2)

	工程	実際の画面	時間[s]														
所要時間	1	0,1 w0	31	31													
	2	2,w1	416		416												
	3	3															
	4	4,w2,w3	413			413											
	5	5,w4,w5	899				475						424				
	6	6,w6,w7	1977					855						1122			
	7	7,w8	148						128						20		
	8	8,w8,w9	160							57						103	
	9	9,w10	382								243						139
		w11	6									6					
作業時間	1	0,1 w0	31	31													
(待ち時間除く)	2	2,w1	5		5												
	3	3															
	4	4,w2,w3	321			321											
	5	5,w4,w5	832				440						392				
	6	6,w6,w7	1279					783						496			
	7	7,w8	139						128						11		
	8	8,w8,w9	54							25						29	
	9	9,w10	382								243						139
		w11	6									6					
入力時間	1	0,1 w0	1	1													
	2	2,w1	0		0												
	3	3	0														
	4	4,w2,w3	0			0											
	5	5,w4,w5	53				15						38				
	6	6,w6,w7	194					61						133			
	7	7,w8	0						0						0		
	8	8,w8,w9	0							0						0	
	9	9,w10	0								0						0
		w11	0									0					
入力回数	1	0,1 w0	0														
	2	2,w1	0														
	3	3	0														
	4	4,w2,w3	0														
	5	5,w4,w5	0														
	6	6,w6,w7	0														
	7	7,w8	0														
	8	8,w8,w9	0														
	9	9,w10	0														
		w11	0														
1サイクル	1		2624														
(待ち時間、TAの説明含む)	2		1808														
			4432														
待ち時間	1	0,1 w0	0	0													
	2	2,w1	411		411												
	3	3															
	4	4,w2,w3	92			92											
	5	5,w4,w5	67				35						32				
	6	6,w6,w7	698					72						626			
	7	7,w8	9						0						9		
	8	8,w8,w9	106							32						74	
	9	9,w10	0								0						0
		w11	0									0					
スケーリング回数				26'35		51'40		260	56								
				28'50		53'00		138	75								
				30'40		54'15		116	77								
				31'45		56'10		62	116								
						63'00			406								
						65'45			167								
						69'05			201								

Fig. 3.4: 集計結果 (GUI1)

| | 工程 | 実際の画面 | 時間[s] | | |

	工程	実際の画面	時間[s]													
所要時間	1	0,1 w0														
	2	2,w1	369	369												
	3	3														
	4	4,w2,w3	180		180											
	5	5,w4,w5	794			491						303				
	6	6,w6,w7	1550				1174						376			
	7	7,w8	64					43						21		
	8	8,w8,w9	195						90						105	
	9	9,w10	263							233						30
		w11	3								3					
作業時間	1	0,1 w0														
(待ち時間除く)	2	2,w1	0	0												
	3	3														
	4	4,w2,w3	153		153											
	5	5,w4,w5	598			416						182				
	6	6,w6,w7	893				688						205			
	7	7,w8	64					43						21		
	8	8,w8,w9	119						44						75	
	9	9,w10	261							231						30
		w11	3								3					
入力時間	1	0,1 w0														
	2	2,w1	0	0												
	3	3														
	4	4,w2,w3	0		0											
	5	5,w4,w5	65			53						12				
	6	6,w6,w7	143				85						58			
	7	7,w8	0					0						0		
	8	8,w8,w9	0						0						0	
	9	9,w10	0							0						0
		w11	0								0					
入力回数	1	0,1 w0	0													
	2	2,w1	0													
	3	3	0													
	4	4,w2,w3	0													
	5	5,w4,w5	0													
	6	6,w6,w7	0													
	7	7,w8	0													
	8	8,w8,w9	0													
	9	9,w10	0													
		w11	0													
1サイクル	1		2583													
(待ち時間、TAの説明含む)	2		835													
			3418													
待ち時間	1	0,1 w0														
	2	2,w1	369	369												
	3	3														
	4	4,w2,w3	27		27											
	5	5,w4,w5	196			75						121				
	6	6,w6,w7	657				486						171			
	7	7,w8	0					0						0		
	8	8,w8,w9	76						46						30	
	9	9,w10	2							2						0
		w11	0								0					
スケーリング回数				18'40		49'40		67	94							
				22'30		50'30		232	53							
				26'50		51'30		261	56							
				29'20		53'00		148	91							
				33'45		54'00		264	63							
				35'20				93								
				36'30				74								

Fig. 3.6: 集計結果 (GUI3)

作業時間	1	2	3	4	5
周波数応答実験	818				
ノミナルプラント作成	136				
重み関数設定	673		186		
コントローラ設計	1560		154		
性能予測	146		61		
消音実験	253	152	118		
性能評価	0	370	200		
その他(準備、データ保存など)	0	246	0		
合計	3586	768	719		

入力時間	1	2	3
周波数応答実験	220		
ノミナルプラント作成	5		
重み関数設定	40		22
コントローラ設計	159		25
性能予測	16		14
消音実験	28	59	37
性能評価		32	28
その他(準備、データ保存など)	0	26	0
合計	468	117	126

待ち時間	1	2	3	4	5
周波数応答実験	1057				
ノミナルプラント作成	11				
重み関数設定	15		10		
コントローラ設計	87		40		
性能予測	3		2		
消音実験	0	10	0		
性能評価		6	0		
その他(準備、データ保存など)	0	0	0		
合計	1173	16	52		

所要時間	1	2	3
周波数応答実験	1875		
ノミナルプラント作成	147		
重み関数設定	688		196
コントローラ設計	1647		194
性能予測	149		63
消音実験	253	162	118
性能評価		376	204
その他(準備、データ保存など)	0	246	0
合計	4759	784	775

所要時間	5846	784	775	合計	7405

重み		
所要時間	688	196
作業時間	673	186
Emacs	80	58
	593	128

比較用

作業時間	1	2	3	4	5
周波数応答実験	409				
ノミナルプラント作成	136				
重み関数設定	673	186			
コントローラ設計	1560	154			
性能予測	146	61			
消音実験	253	118			
性能評価	370	200			
その他(準備、データ保存など)	94	0			
合計	3641	719		4360	

所要時間	1	2	3
周波数応答実験	937.5		
ノミナルプラント作成	147		
重み関数設定	688	196	
コントローラ設計	1647	194	
性能予測	149	63	
消音実験	253	118	
性能評価	376	204	
その他(準備、データ保存など)	94	0	
合計	4291.5	775	
			5066.5

作業時間内訳	CUI1	CUI2	GUI1	GUI2	GUI3
ノミナル	136	54	321	218	153
入力	5	4			
図	63	24	83	52	33
それ以外	68	26	238	166	120
性能予測	146	89	128	151	43
入力	16	16			
図	75	30	52	26	29
その他	55	43	76	125	14
性能評価	370		243	67	231
入力	32				
図	253		164	59	119
その他	85		79	8	112
重み	673	635	440	280	416
入力	40	42	15	10	53
図	191	249	78	242	157
その他	442	344	347	28	206
重みEmacs			コントローラEmacs		
Emacs	80	55	E合計	428	25
作業-E	593	131	作業-E	1043	85

Table. 3.1: 集計まとめ (CUI1)

作業時間	1	2	3	4	5
周波数応答実験	378				
ノミナルプラント作成	54				
重み関数設定	635		454		
コントローラ設計	276	428	289		
性能予測	89				
消音実験	270	119	173		
性能評価					
その他(準備、データ保存等)	245	0	0		
合計	1947	547	916		

入力時間	1	2	3
周波数応答実験	152		
ノミナルプラント作成	4		
重み関数設定	42		30
コントローラ設計	48	99	53
性能予測	16	0	
消音実験	69	54	60
性能評価			
その他(準備、データ保存等)	79	0	0
合計	394	153	143

待ち時間	1	2	3	4	5
周波数応答実験	877				
ノミナルプラント作成	16				
重み関数設定	68		17		
コントローラ設計	28	137	77		
性能予測	4				
消音実験	0	0	0		
性能評価					
その他(準備、データ保存等)	0	0	0		
合計	993	137	94		
所要時間	3975	910	1010		5895

所要時間	1	2	3
周波数応答実験	1255		
ノミナルプラント作成	70		
重み関数設定	703		471
コントローラ設計	304	654	366
性能予測	93	31	
消音実験	270	225	173
性能評価			
その他(準備、データ保存等)	245	0	0
合計	2847	879	1010

比較用

作業時間	1	2	3	4	5
周波数応答実験	189				
ノミナルプラント作成	54				
重み関数設定	635	454			
コントローラ設計	704	289			
性能予測	89				
消音実験	389	173			
性能評価					
その他(準備、データ保存等)	122.5	0			
	2182.5	916		3098.5	
	36'22				

ノミナル	54	重みEmacs		
入力	4	重み	635	454
図	24	Emacs	72	46
その他	26	重み-E	563	408
性能予測	89			
入力	16	コントローラEmacs		
図	30	Emacs合計	143	30
その他	43	作業-E	450	301
性能評価			0	0
入力		Emacs	93	12
図			10	10
その他			5	8
重み	635		15	
入力	42		2	
図	249		5	
その他	344		8	
Emacs	55		5	

Table. 3.2: 集計まとめ (CUI2)

作業時間	1	2	3	4	5
周波数応答実験	5				
ノミナルプラント作成	321				
重み関数設定	440	392			
コントローラ設計	783	507			
性能予測	128				
消音実験	25	168			
性能評価	243				
その他(準備、データ保存など)	37	0			
合計	1982	1067		3049	
	33'02				

入力時間	1	2	3
周波数応答実験	0		
ノミナルプラント作成	0		
重み関数設定	15	38	
コントローラ設計	61	133	
性能予測	0	0	
消音実験	0	0	
性能評価	0	0	
その他(準備、データ保存など)	1	0	
合計	77	171	

待ち時間	1	2	3	4	5
周波数応答実験	411				
ノミナルプラント作成	92				
重み関数設定	35	32			
コントローラ設計	72	626			
性能予測	0	9			
消音実験	0	74			
性能評価	32	0			
その他(準備、データ保存など)	0	0			
合計	642	741			

所要時間	1	2	3
周波数応答実験	416		
ノミナルプラント作成	413		
重み関数設定	475	424	
コントローラ設計	855	1122	
性能予測	128	20	
消音実験	57	103	
性能評価	243	139	
その他(準備、データ保存など)	37	0	
合計	2624	1808	

所要時間(準備等含む)	2624	1808	4432

作業時間内訳	
ノミナル	321
図	83
それ以外	238
性能予測	128
図	52
その他	76
性能評価	243
図	164
その他	79
重み	440
入力	15
図	78
その他	347

Table. 3.3: 集計まとめ (GUI1)

作業時間	1	2	3	4	5	
周波数応答実験	0					
ノミナルプラント作成	218					
重み関数設定	280	192	20	23	31	
コントローラ設計	450	42	122	211	72	
性能予測	151	28	28	24	24	
消音実験	57	34	30	47	41	
性能評価	67	61	24	39	52	
その他	3	2	3	6	0	
合計	1226	359	227	350	220	2382

入力時間	1	2	3	4	5
周波数応答実験	0				
ノミナルプラント作成	0				
重み関数設定	10	22	3	1	7
コントローラ設計	29	5	34	51	25
性能予測	0	0	0	0	0
消音実験	0	0	0	0	0
性能評価	0	0	0	0	0
その他	0	0	0	0	0
合計	39	27	37	52	32

待ち時間	1	2	3	4	5
周波数応答実験	335				
ノミナルプラント作成	0				
重み関数設定	35	32	16	12	10
コントローラ設計	57	25	81	113	73
性能予測	0	0	0	0	0
消音実験	16	16	17	13	16
性能評価	0	0	0	0	0
その他	0	0	0	0	0
合計	443	73	114	138	99

所要時間	1	2	3	4	5
周波数応答実験	335				
ノミナルプラント作成	218				
重み関数設定	315	224	36	35	41
コントローラ設計	507	67	203	324	145
性能予測	151	28	28	24	24
消音実験	73	50	47	60	57
性能評価	67	61	24	39	52
その他	3	2	3	6	0
合計	1669	432	341	488	319

所要時間	1669	432	363	548	319	
合計	1669	432	363	548	319	
合計	3338	432	363	548	319	5000

作業時間内訳	
ノミナル	218
図	52
それ以外	166
性能予測	151
図	26
その他	125
性能評価	67
図	59
その他	8
重み	280
入力	10
図	242
その他	28

Table. 3.4: 集計まとめ (GUI2)

作業時間	1	2	3	4	5
周波数応答実験	0				
ノミナルプラント作成	153				
重み関数設定	416	182			
コントローラ設計	688	205			
性能予測	43	21			
消音実験	44	75			
性能評価	231	30			
その他	3	0			
合計	1578	513		2091	

入力時間	1	2	3
周波数応答実験	0		
ノミナルプラント作成	0		
重み関数設定	53	12	
コントローラ設計	85	58	
性能予測	0	0	
消音実験	0	0	
性能評価	0	0	
その他	0	0	
合計	138	70	

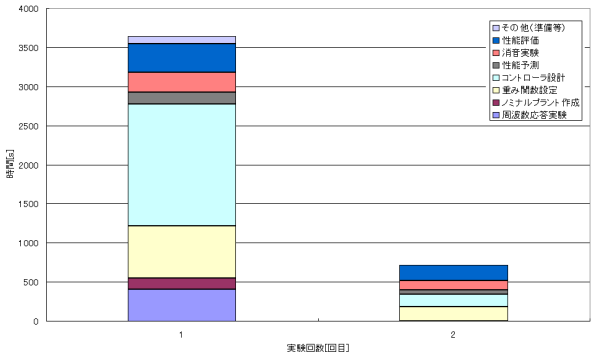
待ち時間	1	2	3	4	5
周波数応答実験	369				
ノミナルプラント作成	27				
重み関数設定	75	121			
コントローラ設計	486	171			
性能予測	0	0			
消音実験	46	30			
性能評価	2	0			
その他	0	0			
合計	1005	322			

所要時間	1	2	3
周波数応答実験	369		
ノミナルプラント作成	180		
重み関数設定	491	303	
コントローラ設計	1174	376	
性能予測	43	21	
消音実験	90	105	
性能評価	233	30	
その他	3	0	
合計	2583	835	

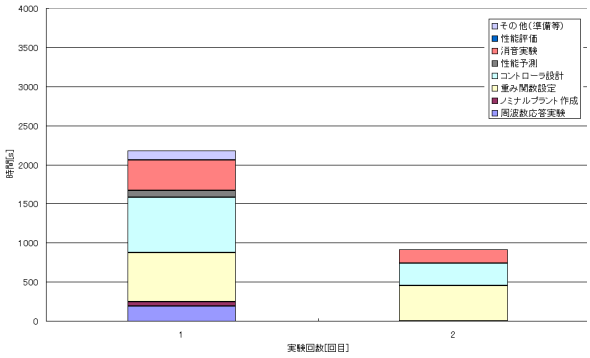
所要時間					
	2583	835		3418	

ノミナル	153
図	33
それ以外	120
性能予測	43
図	29
その他	14
性能評価	231
図	119
その他	112
重み	416
入力	53
図	157
その他	206

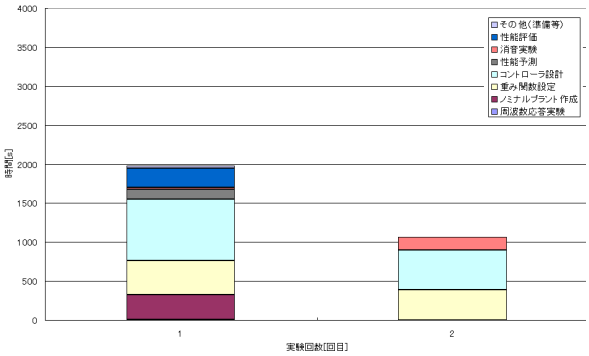
Table. 3.5: 集計まとめ (GUI3)



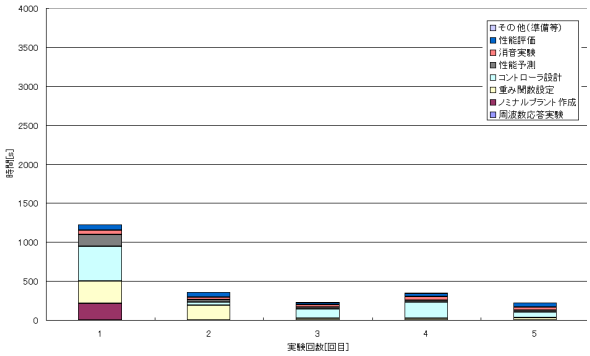
(a)CUI1



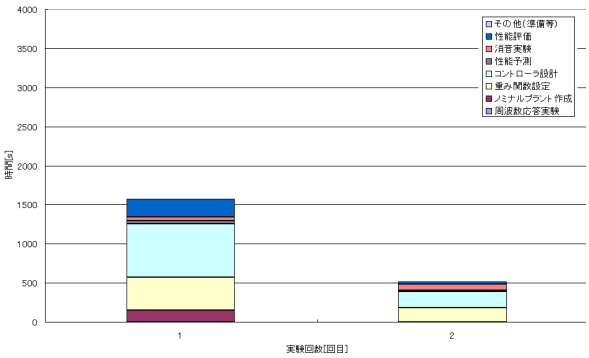
(b)CUI2



(c)GUI1



(d)GUI2



(e)GUI3

Fig. 3.7: 作業時間の比較

2 回目以降は、グループによって作業内容が異なるため、1 回目のみを比較の対象とする。図 3.7 の 1 回目を見ると、全体的に GUI のほうが短くなっていることがわかる。

3.2.1 実験を行う工程

図 3.7 を具体的に工程別に見てみると、短くなっている工程は、周波数応答実験、重み関数設定、消音実験の 3 つである。特に周波数応答実験、消音実験は、ほとんど作業時間が無くなっている。ここでの GUI の作業は 1 回のクリックのみなので、コマンドラインとの差が大きく現れている工程と言える。

3.2.2 図の表示のみを行う工程

短くなっていない工程はノミナルプラント作成、コントローラ設計、性能予測、性能評価の 4 つである。コントローラ設計を除いた 3 つは、図の表示のみを行う工程である。これらを工程別に比較する。

図 3.8 はノミナルプラント作成における作業時間の内訳の比較である。GUI は入力時間がなくなり、その他（主に説明のページ）の時間が長くなっていることがわかる。CUI では、説明ページを見てもコマンドだけに目がいき、コマンドが無ければ次にいってしまうというような、機械的に作業を行っていることが推測される。

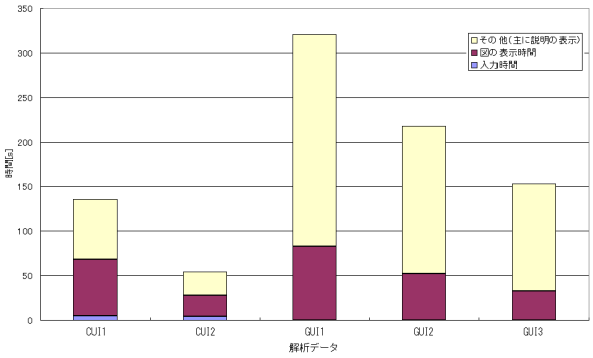


Fig. 3.8: 作業時間内訳（ノミナルプラント作成）

図 3.9 は消音制御系の性能予測における作業時間の内訳の比較である。GUI では入力時間がなくなっていることがわかる。図の表示時間、その他の時間については明確な違いは見られなかった。

図 3.10 は消音制御系の性能評価における作業時間の内訳の比較である。CUI2 はビデオカメラの前に人が立ってしまい記録できなかった。GUI1 では入力がないことがわかるが、データ不足のため図の表示時間、その他の時間については明確な違いはわからなかった。

性能予測、性能評価は、消音制御実験の前後の工程であり、消音制御実験をしてしまえばおおまかな性能がわかってしまう。そのためユーザーにあまり重要視されなかったのかもしれない。

以上 3 つの図の表示を行う工程で、作業時間は短縮されるとは限らないが、入力時間は無くなっている。また出てくる図や、説明のページは同じものを見ているので、時間が短縮されなかった原因は、はっきりとはわからないが、作業が簡単になった分考察時間が増加した可能性がある。

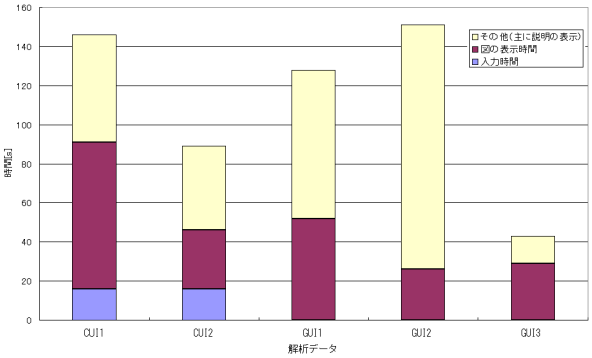


Fig. 3.9: 作業時間内訳（消音制御系の性能予測）

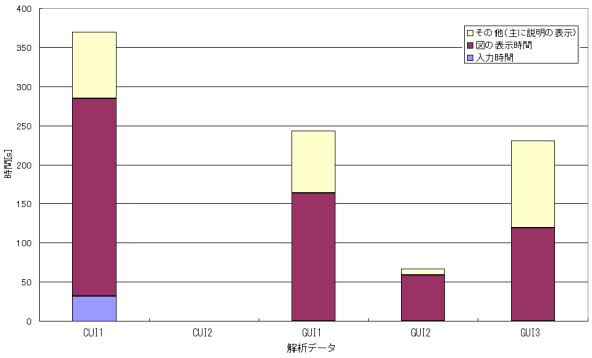


Fig. 3.10: 作業時間内訳（消音制御系の性能評価）

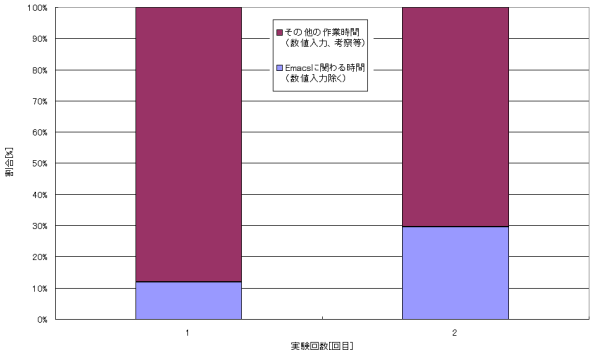
3.2.3 数値入力を行う工程

数値を入力し試行錯誤する、重み関数の設定とコントローラの設計について検証する。この 2 つの工程（特にコントローラの設計）は作業を何度か繰り返すので時間の比較が難しい。そのため、ここではファイル編集に用いられる XEmacs の作業時間における割合（数値入力を除く）を検証する。

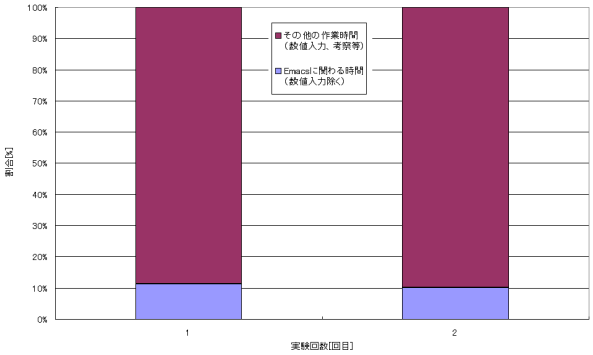
図 3.11 は、CUI の、重み関数設定の作業時間における XEmacs の割合である。1 割から 3 割を XEmacs が占めていることがわかる。

同様に、図 3.12 は、CUI の、コントローラ設計の作業時間における XEmacs の割合である。1 割以上を XEmacs が占めていることがわかる。

図 3.7 を見ると重み関数の設定は平均的に短くなっている。重み関数の設定の「図 2.4 を見ながら、全ての点を覆いつつ、なるべく小さい定数を与える」という作業が、それほど難しくなく、何度も繰り返す必要がなかったということが推測される。そのため、作業が楽な GUI で時間が短縮されたと考えられる。

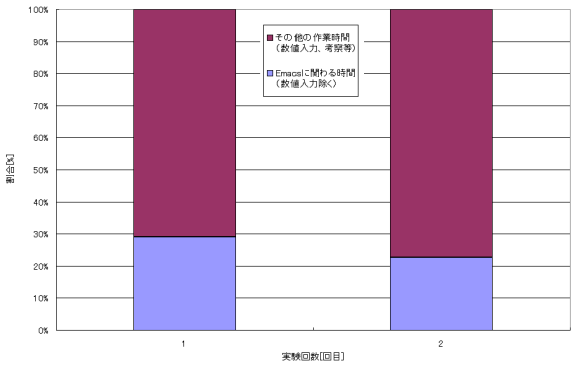


(a)CUI1

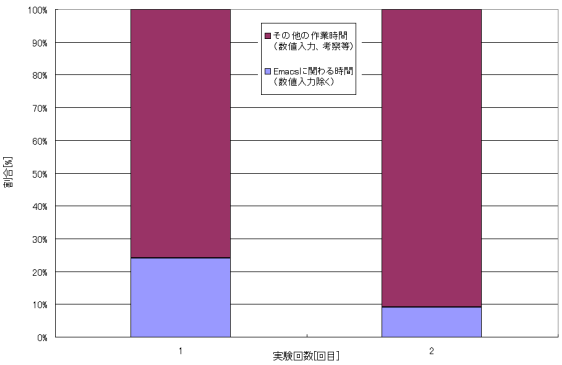


(b)CUI2

Fig. 3.11: 重み関数設定における XEmacs の割合



(a)CUI1



(b)CUI2

Fig. 3.12: コントローラ設計における XEmacs の割合

3.2.4 会話による比較

会話もグループによって数や長さが違ったり、また聞き取れない部分も多々あるため、CUI 操作に関する会話がどれだけ占めているかに注目して、集計を行った。表 3.6,3.7,3.8, 3.9,3.10 は会話の集計結果である。元のデータである会話の記録は付録に記載する

CUI1

1回目	総数	操作	実験	その他	不明
周波数応答	18	4	9	3	2
ノミナルプラント	38	20	8	9	1
重み関数	41	0	25	14	2
コントローラ	215	85	111	17	2
性能予測	28	0	21	7	0
消音実験	15			15	0
性能評価	14		13	0	1

2回目	総数	操作	実験	その他	不明
周波数応答					
ノミナルプラント					
重み関数	60	20	33	7	0
コントローラ	47	3	39	5	0
性能予測	20	0	20	0	0
消音実験	15	1	7	7	0
性能評価	21	0	17	4	0

CUI2

1回目	総数	操作	実験	その他	不明
周波数応答					
ノミナルプラント					
重み関数	50	14	28	7	1
コントローラ	44		37	6	1
性能予測					
消音実験	19	3	2	12	2
性能評価					

GUI1

1回目	総数	操作	実験	その他	不明
周波数応答	0			0	0
ノミナルプラント	33		27	6	0
重み関数	70		60	10	0
コントローラ	130		105	22	3
性能予測	18		14	4	0
消音実験	2		2	0	0
性能評価	15		12	3	0

GUI2

1回目	総数	操作	実験	その他	不明
周波数応答	1			1	0
ノミナルプラント	2			0	2
重み関数	0			0	
コントローラ	3			0	3
性能予測	0			0	
消音実験	3		3	0	
性能評価	0			0	

GUI3

1回目	総数	操作	実験	その他	不明
周波数応答	0			0	
ノミナルプラント	1			0	1
重み関数	4		0	3	1
コントローラ	59		30	24	5
性能予測	3		3	0	0
消音実験	3		3	0	0
性能評価	4			4	0

合計

	操作	実験	その他	不明
CUI1	109	187	65	8
CUI2	17	67	25	4
GUI1	0	220	45	3
GUI2	0	3	1	5
GUI3	0	36	31	7

Table. 3.6: 会話集計 (CUI1)

1回目	総数	操作	実験	その他	不明
周波数応答	6		6	0	0
ノミナルプラント	3		2	1	0
重み関数	13	5	8	0	0
コントローラ	36	0	29	1	6
性能予測	11	0	4	6	1
消音実験	18	4	11	2	1
性能評価					0
2回目	総数	操作	実験	その他	不明
周波数応答					
ノミナルプラント					
重み関数	50	14	28	7	1
コントローラ	44		37	6	1
性能予測				0	0
消音実験	19	3	2	12	2
性能評価				0	0

コントローラ1	9		5	1	3
性能予測1	11		4	6	1
消音実験1	13	4	9	0	0
				0	
コントローラ2	27		24	0	3
性能予測2				0	0
消音実験2	5		2	2	1

Table. 3.7: 会話集計 (CUI2)

1回目	総数	操作	実験	その他	不明
周波数応答	0			0	0
ノミナルプラント	33		27	6	0
重み関数	70		60	10	0
コントローラ	130		105	22	3
性能予測	18		14	4	0
消音実験	2		2	0	0
性能評価	15		12	3	0
2回目	総数	操作	実験	その他	不明
周波数応答				0	
ノミナルプラント				0	
重み関数	40		40	0	0
コントローラ	185		141	43	1
性能予測	8		8	0	0
消音実験	9		7	0	2
性能評価	21		19	2	0

Table. 3.8: 会話集計 (GUI1)

1回目	総数	操作	実験	その他	不明
周波数応答	1			1	0
ノミナルプラント	2			0	2
重み関数	0			0	
コントローラ	3			0	3
性能予測	0			0	
消音実験	3		3	0	
性能評価	0			0	

2回目	総数	操作	実験	その他	不明
周波数応答					
ノミナルプラント					
重み関数	22		21	0	1
コントローラ	2		1	0	1
性能予測	6		6	0	0
消音実験	4		3	0	1
性能評価	1		1	0	

3回目	総数	操作	実験	その他	不明
周波数応答					
ノミナルプラント					
重み関数	0			0	
コントローラ	2		2	0	
性能予測	0			0	
消音実験	1		1	0	
性能評価	5		3	0	2

4回目	総数	操作	実験	その他	不明
周波数応答					
ノミナルプラント					
重み関数	7		6	0	1
コントローラ	18		9	0	9
性能予測	32		32	0	
消音実験	7		6	0	1
性能評価	8		5	0	3

5回目	総数	操作	実験	その他	不明
周波数応答					
ノミナルプラント					
重み関数				0	
コントローラ				0	
性能予測				0	
消音実験				0	
性能評価				0	

Table. 3.9: 会話集計 (GUI2)

1回目	総数	操作	実験	その他	不明
周波数応答	0			0	
ノミナルプラント	1			0	1
重み関数	4		0	3	1
コントローラ	59		30	24	5
性能予測	3		3	0	0
消音実験	3		3	0	0
性能評価	4			4	0
2回目	総数	操作	実験	その他	不明
周波数応答					
ノミナルプラント					
重み関数	32		27	5	0
コントローラ	26		24	2	0
性能予測	2		2	0	
消音実験	0			0	
性能評価	4		4	0	0

Table. 3.10: 会話集計 (GUI3)

図 3.13 は各解析データの会話内容の割合を比較したものである。CUI では 1〜3 割がツールの操作に関する会話で占めていることがわかる。

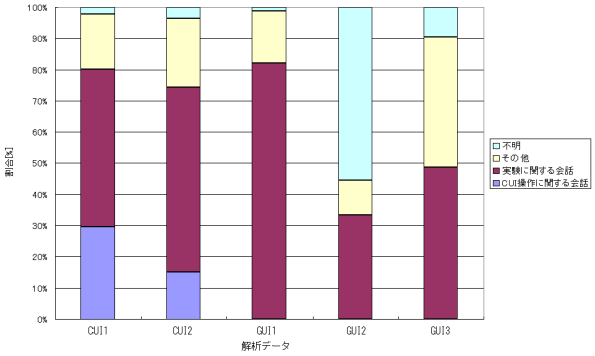


Fig. 3.13: 会話内容の割合

3.2.5 まとめ

以上から、以下のことがわかった。

- GUI は CUI より作業 (キーボード入力) が少ない。
- GUI では実験を行う工程において、作業時間が短縮される。
- CUI では、実験そのものに関係のないツールの操作が負担になっている。

このことから、GUI は CUI よりも作業が楽になるので、教材として有効的であることがわかった。

第 4 章

GUI の改善

4.1 現状システムの改善点

図 4.1 は目標となるシステムの例である。

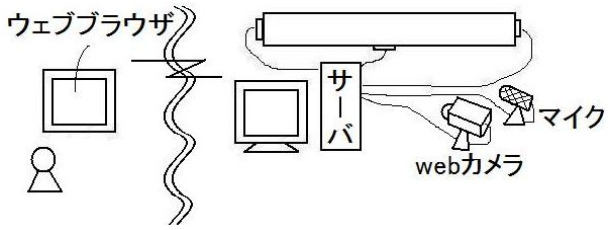


Fig. 4.1: 目標となるシステム例

現状のシステム (GUI) を図 4.1 のように遠隔化するためには、以下のような改善すべき点がある。

- 臨場感の不足

  遠隔実験では、実験装置が目の前にないため、実験をしている感触が得られない。そのため図 4.1 のように、Web カメラやマイクを用いるなどして、臨場感を出すことが必要である。特に本システムでは音を扱っているため、遠隔地から音を聞くことができないのは重要な問題である。

- ユーザー間でのコミュニケーション

  実験に対する考察や意見、質問などがある時、管理者に対してはメールなどでやりとりを行える。しかし、ユーザー同士でのやりとりを行う環境は整っていない。そのため掲示板を設置するなどして、ユーザー間でコミュニケーションを取れるようにする必要がある。

- 割り込み処理

  あるユーザーが実験中に別のユーザーが実験を行おうとした場合、現状では「他の人が実験中です」と出てくるようになっているが、いつ終わったかがわからない。割り込みに対して待ち時間を表示するなどの配慮が必要である。

- 設定可能パラメータの追加

  現状では、設定可能なパラメータは重み関数、スケーリングの 2 つである。実験の自由度を増すため、設定可能パラメータを追加することが必要である。

- その他使いやすさの向上

  ユーザーが、より理解しやすい教材にすることや、操作を簡易にするような工夫は常に必要である。例として説明ページが挙げられる。、現状での説明のページは CUI,GUI ともに同じものであるが、コマンドの説明は GUI には不要であることから、新たに GUI 用のわかりやすい説明ページが必要がある。

本論文では上記の改善項目から、臨場感を得られるように以下の改善を行った。

4.2 web カメラの設置

周波数応答実験には 5 分以上の時間を要する。現状の GUI 教材では、ブラウザの画面上にボタンが一つあるだけである。しかし、クリックしてから何の変化もないため、実験が正しく行われているかどうかわからない。それを改善するため、web カメラを設置した。

web カメラは Axis 221 Network camera を使用し、オシロスコープの静止画像を実験サーバへ FTP 転送するように設定した。オシロスコープでは、z,y,u,w4 つの波形を映し出し、ユーザーは波形が変化していることで、実験装置の動作を確認できる。サーバでは転送された画像を 3 秒おきに更新するページをつくり、周波数応答実験のページに iframe で組み込んだ。(図 4.2)

web カメラに直接アクセスすれば、リアルタイムの動画を見ることができ、その映像を html ファイルに組み込めば、リアルタイムでの映像を見ることが可能だが、ブラウザにアドオンを追加する必要がある、PC の動作が重くなる、web カメラの設定が変更される可能性があるといった問題があるため、静止画像を利用することにした。

(a) 現状

(b) 改善後

Fig. 4.2: ブラウザの画面（周波数応答実験）

4.3 音声 dat ファイルの wav 化

遠隔地では音を聞くことができないため、現状のシステムでは、消音実験の結果をグラフでしか確認することができなかった。そこで音声 dat ファイルを wav 化するスクリプトを用いて、消音実験結果を web 上で聞けるようにした。
消音実験結果（result.dat）のデータファイルから音声データを取り出して wav 形式に変換する。すべてのグラフを表示しているページで、変換したデータを再生できるようにした。またブラウザによっては再生が不可能になることもあるので、データをダウンロードできるようにした。（図 4.3）

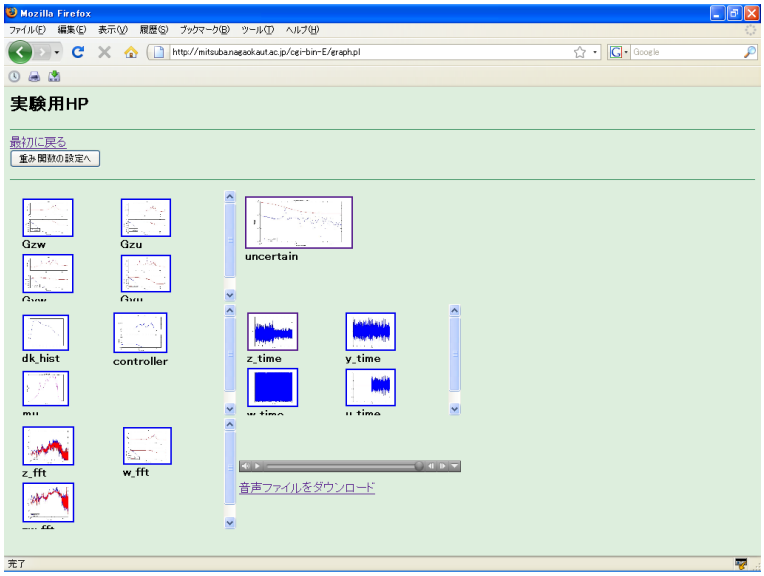


Fig. 4.3: ブラウザの画面（結果閲覧のページ）

第 5 章

結言

本論文では、CUI,GUI,2 つのシステムを比較し、また GUI ベースの教材の改善を行った。CUI と GUI を比較し以下のことが明らかになった。

- キーボード入力が無くなり、実験工程において作業時間が短縮される
- ブラウザに 1 本化することで、ツール (特に XEmacs) による負担が無くなる

以上から、GUI のほうが優れたシステムであることが明らかになり、これを遠隔化する際、臨場感を得るための工夫として web カメラの設置、音声 dat ファイルの wav 化に取り組み、以下のことが可能になった。

- 実験工程において実験装置の動作を可視化
- 消音制御実験結果を聞けるようになった

上記のように使用するツールをブラウザに 1 本化することは、負担がなくなることに加え、余計なものを用意しなくていいという遠隔化の目的に一致するため、大きな効果があると言える。
以上のことから、目標であるシステムの遠隔化に近づいたといえる。

第 6 章

今後の展望

今回の CUI と GUI の比較では、母数が少なく十分な比較が行われたとは言い難い。今後、遠隔実験環境を完成させ、評価を行う際には、なるべく多くの人に利用してもらい十分な比較・検討を行うべきである。

また、システムの遠隔化のために第 4 章で述べた改善点に取り組まなければならない。

本論文では、複雑な制御対称を扱うことが目的であるが、ただ難解な実験にするのではなく、実験環境だけを提供し、ユーザーの好きなように実験をおこなえるようなものが理想である。

謝辞

本研究を進めるにあたり、熱心な指導を頂いた指導教官である小林泰秀准教授、web カメラ等の機材の貸し出し、ビデオテープのデータ化、また多くのアドバイス、指導を頂きました、中平勝子助教に感謝いたします。この場を借りてお礼申し上げます。

参考文献

[1] 大山恭弘, 余錦華, 平松和也, 横田祥　システム制御教育を支援するリモートラボラトリ (2006)

[2] アームロボット遠隔制御学習システム, http://www.teu.ac.jp/ohshe/telelab2/

[3] 菊池達也　小型モータ制御実習用遠隔実験システムの開発 (2000)

付録 A

元データ（解析シート）

付録 B

解析メモ

付録 C

元データ（会話の記録）