

N_013

Linux のリアルタイム制御パッケージを用いた実験環境構築システム Experimental Environment Construction System using Real-Time Control Package on Linux

古賀 雅伸†
Masanobu Koga

岸田 和也†
Kazuya Kishida

1. はじめに

制御系の設計プロセスは、モデリング、制御対象の解析、制御器設計、シミュレーション、制御実験という流れで行われる。このうち、モデリングからシミュレーションまでのプロセスについては、数多くの解決策やツールが提案・開発されている [1][2]。そのため、教員は、それらツールを講義や演習等で活用することにより、学生にモデリングからシミュレーションまでの制御系設計プロセスについて容易かつ効果的に教授することができる。

しかし、制御実験のプロセスにおいては、支援するツールが少ない。そのため、制御実験の実施には、リアルタイム OS のインストールやメンテナンスなど、制御対象をリアルタイム制御するために実験環境を構築・維持する作業が必要となる。これは、制御実験を実施する教員への大きな負担となる。したがって、制御工学を教授する際は、制御実験を行わずシミュレーションまでで終了するケースが多い。しかし、制御実験は、学生が制御工学に対する理解を更に深めることができる有効な教育手段であることが報告されている [3]。

そこで本研究は、以上の制御実験に関する問題点を考慮し、実験環境の構築・維持に要するリアルタイム OS のインストールやメンテナンスを行うことなく、容易に実験環境を構築するシステムを開発した。

本稿では、本システムを倒立振子の安定化制御実験に適用した事例について述べる。

2. 実験環境構築システム

2.1 システムの特徴

実験環境構築システムの特徴を以下に示す。

1. CD ブートするだけで容易に実験環境を構築できる (図 1 参照)。実験環境の構築に必要なリアルタイム OS のインストールやメンテナンスは不要である。
2. システムの内容を改変することはできない。学生の不用意な操作によりシステムがダメージを受けることはない。
3. CD を複製するだけで大規模な実験環境を容易に構築することができる。

2.2 システムの構成

実験環境構築システムは、図 2 のように CD 起動 Linux である KNOPPIX [4] がシステムのベースとなっている。

2.2.1 KNOPPIX

KNOPPIX は、Debian GNU/Linux をベースにした CD 起動 Linux である。KNOPPIX の特徴を以下に示す。

1. CD ブートするだけで Linux 環境を構築する。ハードディスクへのインストール作業は不要である。
2. オープンソースソフトウェアである。独立行政法人産業技術総合研究所が日本語版 KNOPPIX を開発・公開している。

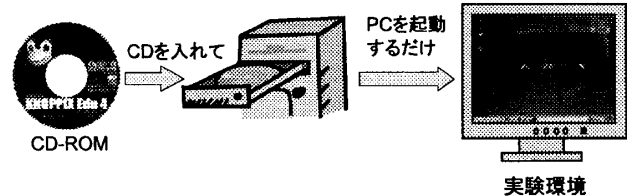


図 1: 実験環境構築システム

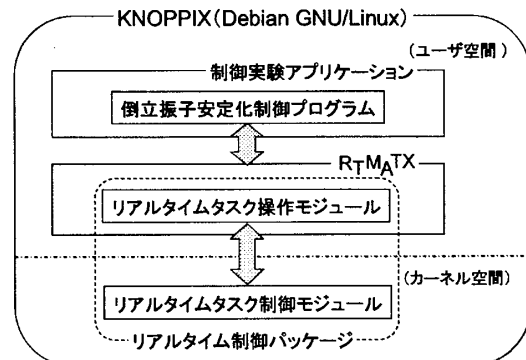


図 2: 実験環境構築システムの構成

3. 特定の用途向けにカスタマイズすることができる。医療や教育分野など、様々な分野で活用されている。

本研究は、KNOPPIX の特徴に着目し、KNOPPIX を制御実験用にカスタマイズすることにより、CD ブートするだけで容易に実験環境を構築するシステムを開発した。本研究がカスタマイズを行った点を以下に示す。

制御実験の実験環境には、制御対象に対するリアルタイム制御を可能にするために、時間制約の保証が必要である。しかし、Linux は、非リアルタイム OS であるため、時間制約を保証することができない。つまり、KNOPPIX 上で制御実験を実施することは困難である。

そこで、本研究はこの問題点に対し、本研究が研究開発しているリアルタイム制御パッケージ [5] を KNOPPIX に組み込むことにより、KNOPPIX 上での制御実験を可能にした。

2.2.2 リアルタイム制御パッケージ

リアルタイム制御パッケージは、Linux カーネル 2.6 上でリアルタイム処理に必要な機能を提供するミドルウェアである。2つのモジュールから構成している。1つ目は、リアルタイム処理に必要な機能を提供するリアルタイムタスク操作モジュールである。表 1 に示す 6 つの関数を提供する。2つ目は、リアルタイムタスクの制御・管理を行うリアルタイムタスク制御モジュールである。カーネルモジュールで構成している。

以下にリアルタイム制御パッケージの特徴を示す。

1. リアルタイムタスク制御モジュールを Linux カーネルに組み込むだけでリアルタイム制御パッケージが利用できる。カーネルの再構築を行う必要はない。
2. リアルタイムタスクがユーザ空間のタスクとして動作する。そのため、タスク上で Linux 既存のソフト

†九州工業大学 情報工学部

表 1: リアルタイムタスク操作モジュール

| 関数名 | 説明 |
|--------------------------|--------------|
| pthread_create_np | RT タスクを作成 |
| pthread_suspend_np | RT タスクの実行を中断 |
| pthread_wakeup_np | RT タスクの実行を再開 |
| pthread_make_periodic_np | 周期処理の設定 |
| pthread_stop_periodic_np | 周期処理を終了 |
| pthread_wait_np | 次の周期まで待つ |

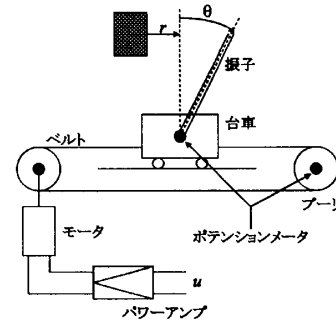


図 4: 制御対象の倒立振り子系

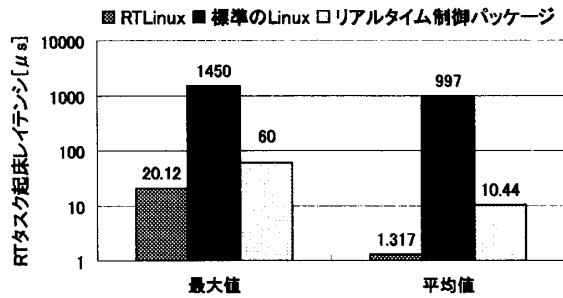


図 3: リアルタイム制御パッケージの精度

ウェア資源を利用することができ、かつメモリ保護される。

リアルタイム制御パッケージの性能は、標準の Linux と Linux をリアルタイム OS に拡張する RTLinux[6] と比較した場合、図 3 に示す結果となる。図 3 は、負荷[†]を与えたシステム上でリアルタイムタスクを 1[ms] 周期で動作させた場合の起床レイテンシの最大値と平均値を表す。標準の Linux の最大値は、RTLinux と比べ約 72 倍になるのに対し、パッケージを用いることで約 3 倍に改善される。同様に平均値は、パッケージを用いることで 957 倍から約 8 倍に改善される。

2.2.3 制御実験アプリケーション

本研究は、図 4 に示す倒立振り子系の安定化制御を行う倒立振り子安定化制御プログラムを KNOPPIX に組み込んでいる。プログラムは、RTMATX[7] で実装している。

3. 評価実験

本実験は、実験環境構築システム上で倒立振り子の安定化制御実験を行うことにより、システムの有効性を示す。実験で使用したマシンのスペックは、CPU が Celeron M 1.30[GHz]、メモリが 512[MB] である。

台車の目標位置を 5[s] 毎、ステップ状に 10[cm] 変動させた場合の倒立振り子系の応答波形を図 5 に示す。実線は実測波形、破線はシミュレーション波形である。図 5 より、実験環境構築システム上における倒立振り子系の応答は、定常状態のシミュレーション波形と比較した場合、振り子の角度が約 1.15[deg]、台車の位置が約 1.34[cm] の誤差範囲で制御される。

4. まとめ

本研究は、CD をブートするだけで容易に制御実験の実験環境を構築するシステムを開発した。このシステムにより、実験環境を構築・維持に要するリアルタイム OS のインストールやメンテナンスを行う必要はない。その

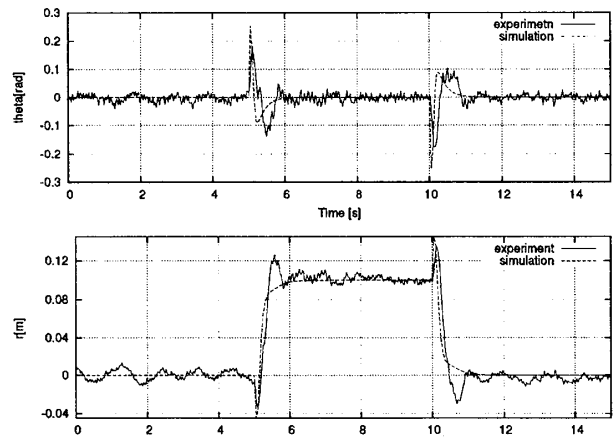


図 5: 倒立振り子の応答波形 (上: 振り子、下: 台車)

ため、制御実験を実施する教員への負担を大幅に軽減することが可能となる。

現在このシステムは、来年度、本大学の倒立振り子の実験環境を構築する際に使用する予定である。今後は、CD だけではなく USB やネットワークなど、システムをブートする形式を増やし、更に容易に実験環境を構築することのできるシステムへ発展させていきたいと考えている。

本研究の一部は、科学研究費補助金(基盤 C:17560396)による助成をうけて行われた。ここに誠意を表す。

参考文献

- [1] MATLAB. <http://www.cybernet.co.jp/matlab>.
- [2] 筒井 勇介. ブロック線図を用いた制御系のモデル化・シミュレーションプラットフォームの開発, 2006. 2005 年度修士学位論文.
- [3] 平田 研二・大原 伸介・新銀 秀徳. 制御系設計・実験統合環境の構築と学部生実験への適用による考察, 2005. 第 5 回計測自動制御学会制御部門大会.
- [4] KNOPPIX Japanese edition <http://unit.aist.go.jp/itri/knoppix/>.
- [5] 古賀 雅伸・岸田 和也. Linux カーネル 2.6 を用いたリアルタイム制御パッケージの開発, 2005. 第 49 回システム制御情報学会.
- [6] FSMLabs. <http://www.fsmlabs.com>.
- [7] 古賀 雅伸. 制御系のシミュレーション環境とリアルタイム制御環境の融合. 電気学会論文誌, 118 巻 4 号.

[†]ディスクアクセスを無限ループする