

接触情報にもとづくマニピュレータの経路自動修正方法の 部品配置タスクにおける利用可能性の検証

○笠原 佑太 Francisco Jesús Arjonilla García 早川 智洋 小林 祐一（静岡大学）
竹下 啓嗣 和田 侑也 中村 陽一郎（芝浦機械）

日本の産業現場では人手不足が問題となっており、人間の代わりに働くロボットが求められている。しかしロボットの動作教示は従来の方法では非常に時間がかかり教示者の負担が大きい。この問題を解決するために、ロボットの動作経路の通過点とパラメータを教示者が設定するだけで、接触時に得られる力情報に基づいて経路が自動的に補正される方法を提案する。本研究では、提案手法を部品配置タスクを想定したシミュレーションを行い、マニピュレータへの適用可能性を検証した。

1. 緒言

生産現場では生産ラインへの部品の供給や配置、荷物の積み下ろしといった生産作業の手助けをする生産支援ロボットの需要が高まりつつある。このようなロボットは環境や作業対象物との間に接触や拘束を伴う作業であり、高度な物体操作を求められる。また、生産現場への導入にあたり現場ではロボットのための環境が整備されていないという問題がある。そのため、人間からロボットへの動作の教示が必要になる。従来の動作教示の方法としては教示者が直接ロボットを操作して動作教示するダイレクトティーチングやモーションプランニングなどの自律型がある。しかし、これらの教示方法は教示者へある程度の技術が求められ教示者への負担が大きいという問題や障害物や対象物の事前環境モデルの構築にコストと時間がかかるという問題がある。以上の背景から、本研究では生産支援ロボット導入時の教示者の負担軽減のためのロボットへの作業教示時間の短縮を目的とする動作経路生成の自動化について考える。

Losey *et al.* はロボットの自律動作中に人間が物理的に介入して動作を修正し、その情報からロボットが最適な動作経路を生成する手法を提案し、リアルタイムに軌道を更新する学習ルールを構築した [1]。しかし、この研究では人間が直接動作教示を行うことやリアルタイムで動作教示を行う必要があり、ロボットへの作業教示時間の短縮の問題はまだ解決されていない。

そこで本研究では、荒井らの研究 [2][3][4][5][6] を先行研究として、生産支援ロボット導入における教示者の負担削減を目的とする。ロボットに実際に教示した経路を追従させて環境との接触時に得られる力覚情報をもとに障害物の位置を推定、障害物に関するコスト関数を定義して最小化問題を解くことで経路修正を自動で行う。本論文では、先行研究の実ロボットタスクでの利用可能性の検証として、配置された糸巻きを取るというタスクをシミュレーションで行う。

2. 問題設定

本研究では3次元6自由度のマニピュレータの手先の位置姿勢の6自由度の動作経路生成問題を扱う。教示者が設定した経路を自動修正し、ロボットが持つ物体が障害物に接触しない経路を生成することを目標とする。経路の設定は、教示者が最初に一度だけ、始点、終点、経由点を設定することで行い、経路曲線はそれ

らをスプライン補間することで得る。経路修正では、環境モデルなどの事前情報を使わず、接触時に得られる手先の位置姿勢とセンサから得られる接触力を用いる。

3. 経路修正方法

3.1 経路曲線の表現方法

経路曲線は手先の位置姿勢の6次元の曲線として表現する。教示者が設定する始点、終点、経由点を制御点と呼ぶこととする。制御点は手先位置 $\mathbf{p} = (x, y, z)$ 、手先姿勢 $\mathbf{r} = (\phi v_x, \phi v_y, \phi v_z)$ として次式のように設定する。ここで、初期姿勢からの回転角を ϕ 、回転軸を $\mathbf{v} = (v_x, v_y, v_z)$ とする。

$$\mathbf{x} = \begin{bmatrix} x & y & z & \phi v_x & \phi v_y & \phi v_z \end{bmatrix}^T \quad (1)$$

経路曲線の形は N 個の制御点 $\mathbf{x} \in \mathbb{R}^6$ の集合 $\mathbf{X} = \{\mathbf{x}_i \mid i = 1, \dots, N\}$ (制御点列) により設定し、制御点列 $\mathbf{X}$ をスプライン補間することにより曲線とする。

3.2 接触情報

接触情報の概要を図1に示す。マニピュレータの手先の接触位置姿勢 $\mathbf{x}^c \in \mathbb{R}^6$ は、マニピュレータの手先の接触位置 $\mathbf{p}^c \in \mathbb{R}^3$ と接触姿勢 $\mathbf{r}^c \in \mathbb{R}^3$ から次のように定義する。

$$\mathbf{x}^c = \begin{bmatrix} \mathbf{p}^c & \mathbf{r}^c \end{bmatrix}^T \quad (2)$$

接触力 $\mathbf{F}^c \in \mathbb{R}^6$ は、マニピュレータの手先が受ける並進力 $\mathbf{f}^c \in \mathbb{R}^3$ と回転力 $\mathbf{n}^c \in \mathbb{R}^3$ から次のように定義する。 $\mathbf{p}^c \in \mathbb{R}^3$ と接触姿勢 $\mathbf{r}^c \in \mathbb{R}^3$ から次のように定義する。

$$\mathbf{F}^c = \begin{bmatrix} \mathbf{f}^c & \mathbf{n}^c \end{bmatrix}^T \quad (3)$$

経路修正で使用する接触情報 $\mathbf{c}$ は、接触時の位置姿勢 $\mathbf{x}^c \in \mathbb{R}^6$ と接触力 $\mathbf{F}^c$ から次のように定義する。

$$\mathbf{c} = (\mathbf{x}^c, \mathbf{F}^c) \quad (4)$$

接触情報 $\mathbf{c}$ は1回の接触で1組得られ、 i 回目の接触情報を $\mathbf{c}_i$ とする。

3.3 経路自動修正の手順

経路生成の手順を図2に示す。まず教示者が初期経路の制御点列 $\mathbf{X}^{\text{init}}$ を設定し、これを追従経路の制御点列 $\mathbf{X}^{\text{follow}}$ として用いる。マニピュレータは、この制

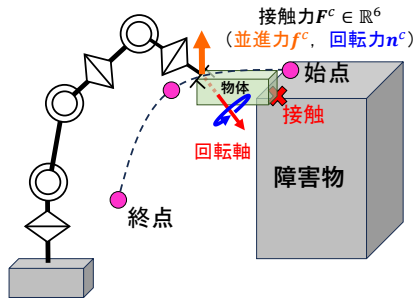


図1 マニピュレータ

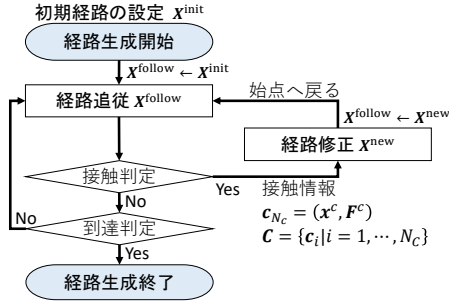


図2 経路生成の手順

御点列 $\mathbf{X}^{\text{follow}}$ から得られる経路を追従し、障害物との接触判定を行う。接触が検出された場合は、その時点の接触情報 $\mathbf{c}_i$ (接触回数 i) を取得し、接触情報履歴 $\mathbf{C}$ に記録する。接触 N_c 回目までに取得された接触情報履歴 $\mathbf{C}$ は次のように表される。

$$\mathbf{C} = \{\mathbf{c}_i \mid i = 1, \dots, N_c\} \quad (5)$$

得られた接触情報履歴 $\mathbf{C}$ を用いて目的関数 L を定義し、最適化計算によって制御点列 $\mathbf{X}$ を更新し、経路を修正する。ただし、始点および終点は固定とし、経由点のみ変更する。本研究では、最適化問題の解法として Nelder-Mead 法 [7] を採用している。

$$\mathbf{X}^* = \arg \min_{\mathbf{X}} L(\mathbf{X}, \mathbf{C}) \quad (6)$$

最適化計算により得られた制御点列 $\mathbf{X}^*$ を新たな追従経路の制御点列 $\mathbf{X}^{\text{follow}}$ として設定する。ロボットを接触位置から始点へ戻した後、再び $\mathbf{X}^{\text{follow}}$ にもとづく経路を追従させる。接触が検出されなければ、目標位置 (経路の終点) への到達判定を行い、到達が確認された時点で経路生成を終了する。

3.4 経路の評価

目的関数 目的関数 L は以下の4項目を評価基準としたコスト関数 $G_i (i = 1, 2, 3, 4)$ で構成される。

- (1) 接触情報による評価
- (2) 初期経路の経由点と経路修正後の経由点との距離と回転量
- (3) 全体の経路長と回転量の総和
- (4) 隣接経由点間の間隔の均等さ

各関数にそれぞれ重み ($\alpha_1, \alpha_2, \alpha_3, \alpha_4$) をかけたものを、目的関数 L とする。目的関数 L を次式に示す。

$$L(\mathbf{X}, \mathbf{C}) = \alpha_1 G_1(\mathbf{X}, \mathbf{C}) + \sum_{i=2}^4 \alpha_i G_i(\mathbf{X}) \quad (7)$$

接触情報を反映させた経路の評価 接触情報を利用して定義したコスト関数 G_1 について説明する。接触情報 $\mathbf{c} = (\mathbf{x}^c, \mathbf{F}^c)$ を用いて障害物位置で高くなるコスト g_1^p と障害物位置周辺で障害物のある方向へ姿勢変化すると高くなるコスト g_1^r を定義し、大きい方をコスト g_1 と定義する。コスト関数 G_1 は g_1 が高い領域を通ると値が大きくなるように定義する。これにより、位置または姿勢を変化させることで障害物との接触の無い経路曲線を生成する。具体的には、経路曲線上に評価点 $\mathbf{x}_k^q (k = 1, \dots, N_q)$ をとり、各評価点におけるコスト g_1 の和をコスト関数 G_1 としている。評価点の集合 $\mathbf{X}^q = \{\mathbf{x}_k^q \in \mathbb{R}^6 \mid k = 1, \dots, N_q\}$ は制御点列 $\mathbf{X}$ をスプライン補間して得られる曲線上の点の集合であるので、制御点列 $\mathbf{X}$ と評価点数 N_q より求められる。ここで、評価点 $\mathbf{x}^q$ の位置の要素を評価位置 $\mathbf{p}^q \in \mathbb{R}^3$ と姿勢の要素を評価姿勢 $\mathbf{r}^q \in \mathbb{R}^3$ とする。

$$G_1(\mathbf{X}, \mathbf{C}) = \sum_{k=1}^{N_q} \max_{i=1, \dots, N_c} g_1(\mathbf{x}_k^q, \mathbf{c}_i) \quad (8)$$

$$g_1(\mathbf{x}_k^q, \mathbf{c}_i) = \max(g_1^p(\mathbf{p}_k^q, \mathbf{c}_i), g_1^r(\mathbf{r}_k^q, \mathbf{c}_i)) \quad (9)$$

始めに評価関数 G_1 の位置に関するコスト g_1^p について説明する。位置に関するコスト g_1^p の例を図3に示す。3のように、コスト g_1^p は接触位置 $\mathbf{p}^c$ を基準として接触力の並進力方向 $\mathbf{f}^c$ と逆向き方向に大きな値をとるように定義する。経路曲線の位置に関して障害物の無い方向へ修正するためのコストとなっている。 a_x, γ_x, σ_x はパラメータ (定数) である。

$$g_1^p(\mathbf{p}^q, \mathbf{c}) = \prod_{i=1}^3 g_{1,i}(\mathbf{p}^q, \mathbf{c}) \quad (10)$$

$$g_{1,1}^p(\mathbf{p}^q, \mathbf{c}) = \frac{1}{1 + \exp[a_x z_1(\mathbf{p}^q | \mathbf{c})]} + \frac{1}{1 + \exp[-a_x z_1(\mathbf{p}^q | \mathbf{c}) + \gamma_x]} - 1 \quad (11)$$

$$g_{1,i}^p(\mathbf{p}^q, \mathbf{c}) = \exp\left[-\frac{z_i^2(\mathbf{p}^q | \mathbf{c})}{2\sigma_x^2}\right], \quad i = 2, 3 \quad (12)$$

ただし、 $z_i(\mathbf{p}^q | \mathbf{c}) (i = 1, 2, 3)$ は次式より得る。 $\perp$ は垂直ベクトルを表す記号である。

$$z_1(\mathbf{p}^q | \mathbf{c}) = \frac{\mathbf{f}}{\|\mathbf{f}\|} \cdot (\mathbf{p}^q - \mathbf{p}^c), \quad (\mathbf{p}^c, \mathbf{f}^c) \in \mathbf{c} \quad (13)$$

$$z_i(\mathbf{p}^q | \mathbf{c}) = \left(\frac{\mathbf{f}}{\|\mathbf{f}\|}\right)^{\perp i} \cdot (\mathbf{p}^q - \mathbf{p}^c),$$

$$(\mathbf{p}^c, \mathbf{f}^c) \in \mathbf{c}, \quad i = 2, 3 \quad (14)$$

次に評価関数 G_1 の姿勢に関するコスト g_1^r について説明する。コスト g_1^r は接触したときの姿勢を基準に障

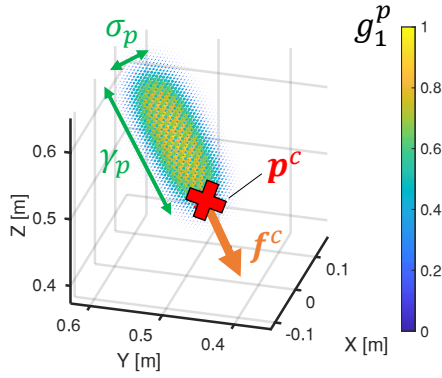


図3 接触情報を利用した位置のコスト

害物がある方向に姿勢が回転すると大きな値をとるように定義する。これにより、接触を回避する方向への姿勢変化（回転）を促すことができる。ここで、接触姿勢 r^c から評価姿勢 r^q への姿勢変化の回転軸 v^q 、回転量 ψ とする。障害物との接触を回避する方向への姿勢変化を促すためのコスト g_1^r を実現するために3つのコストを定義する。1つは評価姿勢の軸に関するコスト h^v 、2つ目は評価姿勢の回転量に関するコスト h^ϕ 、3つ目は接触位置付近での姿勢の変化を促すためのコスト $\hat{g}^p$ である。以下でそれぞれの定義を説明する。まず、評価姿勢の軸を評価するためのコスト h^v の定義を説明する。接触姿勢から障害物との接触を回避する方向に姿勢を回転させるために、望ましい回転軸となるものが必要になる。 h^v は望ましい回転軸で姿勢を回転させるためのコストとなる。ここで、望ましい回転軸 v^c を回転力 n^c を用いて次のように定義する。

$$v^c = \frac{n^c}{\|n^c\|} \quad (15)$$

望ましい回転軸 v^c と姿勢変化の回転軸 v^q の内積が1に近いほどコストが小さくなるようにコスト h^v を次式のように定義する。ただし、 a_v はゲイン（定数）、 b_v はパラメータである。

$$h^v(r^q) = \frac{1}{1 + \exp(a_v(v^c \cdot v^q - \cos b_v))} \quad (16)$$

次に、評価姿勢の回転量を評価するためのコスト h^ϕ の定義を説明する。 h^v で設定した軸を回転軸として姿勢を回転させるために、回転量が大きいほどコストが小さくなるようにコスト h^ϕ を次のように定義する。ただし、 a_ϕ はゲイン（定数）である。

$$h^\phi(r^q) = \frac{1}{1 + \exp(a_\phi \psi)} \quad (17)$$

以上で定義した2つの関数 h^v 、 h^ϕ を用いて、評価姿勢の軸が望ましい回転軸に近いかつ、回転量が大きいときにコストが減少する評価を表現するため、姿勢に関するコスト $\hat{g}^r$ を定義する。接触位置での姿勢に関するコスト $\hat{g}^r$ を図4に示す。

$$\hat{g}^r(r^q) = 1 - (1 - h^v(r^q))(1 - h^\phi(r^q)) \quad (18)$$

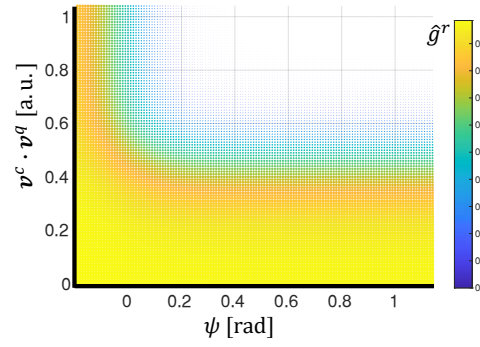


図4 接触情報を利用した姿勢のコスト

ここで、接触位置付近での姿勢変化を促すためのコスト $\hat{g}^p$ を次式のように定義する。ただし、 a_p はゲイン（定数）、 b_p はパラメータである。

$$\hat{g}^p(p^q) = \frac{1}{1 + \exp(-a_p(z_o - b_p))} \quad (19)$$

ただし、軸変数 z_o は以下のとおりとする。

$$z_o = \|p^q - p^c\| \quad (20)$$

以上より、接触位置付近での姿勢変化を促すコスト g_1^r を次式のように定義する。

$$g_1^r(x^q) = \hat{g}^p(p^q) \hat{g}^r(r^q) \quad (21)$$

経路を補正する評価項目 経路の評価基準2～4に基づく、経路の形状を補正するための評価関数 G_2, G_3, G_4 について定義する。また、制御点列 $\mathbf{X}$ は N 個の制御点で与えられるとする。

$$\mathbf{X} = \{x_i | i = 1, \dots, N\} \quad (22)$$

ここで、パラメータ t_p は位置に関する経路の形状を厳しく評価するための重み、パラメータ t_o は姿勢に関する経路の形状を厳しく評価するための重みであり、これらを調整することで位置姿勢の修正量を調節することができる。評価基準2に基づいて評価関数 G_2 を定義する。ここで、初期経路の制御点の姿勢 r_i^{init} から制御点の姿勢 r_i への姿勢変化の回転量を $\Delta\psi_i^{\text{init}}$ とする。

$$G_2(\mathbf{X}) = t_p \sum_{i=1}^N \|p_i - p_i^{\text{init}}\| + t_o \sum_{i=1}^N \Delta\psi_i^{\text{init}} \quad (23)$$

評価基準3に基づいて評価関数 G_3 を定義する。ここで、評価点の姿勢 r_k^q から評価点の姿勢 r_{k+1}^q への姿勢変化の回転量を $\Delta\psi_{k,k+1}$ とする。

$$G_3(\mathbf{X}) = t_p \sum_{k=1}^{N_q-1} \|p_{k+1}^q(\mathbf{X}) - p_k^q(\mathbf{X})\| + t_o \sum_{k=1}^{N_q-1} \Delta\psi_{k,k+1} \quad (24)$$

評価基準4に基づいて評価関数 g_4 を定義する。ここで、制御点の姿勢 r_N から制御点の姿勢 r_{N+1} への姿勢変化の回転量を $\Delta\psi_{N,N+1}$ とする。

$$G_4(\mathbf{X}) = t_p \sum_{i=1}^{N-1} |\Delta p_i - \Delta p_{\text{ave}}| + t_o \sum_{i=1}^{N-1} |\phi_i - \psi_{\text{ave}}| \quad (25)$$

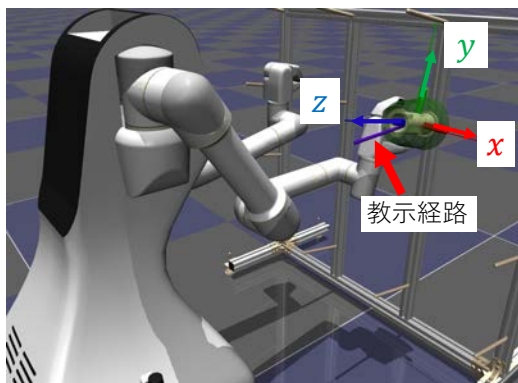


図5 シミュレーション環境と教示経路

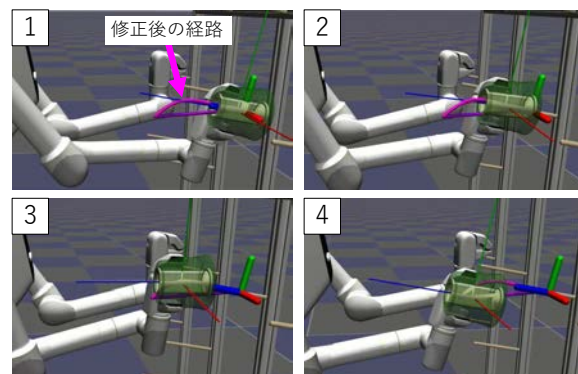


図6 シミュレーション結果

ただし、以下のとおりとする。

$$\Delta p_i = \|p_{i+1} - p_i\| \quad (26)$$

$$\Delta p_{ave} = \frac{1}{N-1} \sum_{j=1}^{N-1} \Delta p_i \quad (27)$$

$$\psi_{ave} = \frac{1}{N-1} \sum_{j=1}^{N-1} \Delta \psi_{N,N+1} \quad (28)$$

4. シミュレーション

4.1 目的と条件

提案手法のマニピュレータへの手先の位置姿勢に関する経路生成方法としての利用可能性の検証を目的として、オープンソースの物理エンジンのMuJoCo[8]を用いて3次元6自由度マニピュレータの動作経路追従シミュレーションを行った。本シミュレーションでは、マニピュレータの手先に力覚センサを設置し、センサ情報より接触判定と接触力の取得を行った。図5に教示経路とシミュレーション環境を示す。図5のように棒に掛けられて配置されている糸巻きを取り外すというタスクを想定してシミュレーションを行う。また、経路修正の計算で用いたパラメータの値を表1に示す。

表1 パラメータ

Parameter	Value	Parameter	Value
α	(1, 5, 5, 4)	b_v	$\frac{\pi}{6}$
a_x	50	a_ϕ	10
γ_x	0.5	t_p	30
σ_x	0.08	t_o	0.01
a_v	10	N_q	101

4.2 結果

シミュレーション結果を図6に示す。本シミュレーションでは、修正回数2回であった。図6のように、教示した経路を修正し、位置姿勢を変化させて障害物を回避する経路を生成できることを確認できた。しかしながら、複数回接触した場合に姿勢を柔軟に変化させることによる障害物の回避ができないことが問題として挙げられる。そのため、姿勢に関するコスト関数を改良する必要がある。

5. 結言

本研究では、生産支援ロボット導入における動作教示者の負担削減を目的とした手先姿勢を考慮した動作経路の自動修正手法の提案と提案手法の利用可能性の検証として設置された糸巻きを取るタスクのシミュレーションを行った。提案手法により、位置姿勢変化による障害物との接触が回避できることが確認された。今後の課題としては、複数回接触したときに柔軟に姿勢を変化させるためのコスト関数の設計方法の改良、提案した経路生成手法に関して調節する必要のあるパラメータを削減させる手法の提案、またはパラメータの自動調整が挙げられる。

参考文献

- [1] D. P. Losey, M. K. O' Malley: "Learning the Correct Robot Trajectory in Real- Time from Physical Human Interaction", ACM Transaction on Human-Robot Interaction, vol.1, no.1, pp.1—8, (2019)
- [2] 荒井康太, 石村芳暉, 小林祐一, 伊部公紀: マニピュレータへの作業教示時間短縮のための力覚情報を用いた経路生成およびそのメタパラメータの自動調整, 第33回自律分散システム・シンポジウム予稿集, 1A1-1 (2021)
- [3] 榎原典佑, 小林祐一, F. J. Arjonilla García: マニピュレータへの動作教示時間短縮のための経路最適化とメタパラメータの自動調整, 第49回知能システムシンポジウム講演資料, C3-3 (2022)
- [4] 高見理瑛, 小林祐一, F. J. Arjonilla García: 単純な動作教示と接触情報にもとづくマニピュレータの経路生成, ロボティクス・メカトロニクス講演会講演概要集 2023, 2A1-E14 (2023)
- [5] 笠原佑太, 高見理瑛, 小林祐一, F. J. Arjonilla García, 中村陽一郎, 竹下啓嗣: 接触情報にもとづく3次元空間における手先姿勢を考慮したマニピュレータの経路生成, ロボティクス・メカトロニクス講演会講演概要集 2024, 2A1-N05 (2024)
- [6] 高見理瑛, 笠原佑太, F. J. Arjonilla García, 早川智洋, 小林祐一, 竹下啓嗣, 和田侑也, 中村陽一郎: マニピュレータへの動作教示作業簡単化のための接触情報にもとづく経路生成, 第52回知能システムシンポジウム予稿集, A1-1, pp.1-6 (2025)
- [7] J. A. Nelder and R. Mead: A simplex method for function minimization, The computer journal, 7 No. 4, 308/313 (1965)
- [8] E. Todorov, T. Erez and Y. Tassa: MuJoCo: A physics engine for model-based control, IEEE/RSJ International Conference on Intelligent Robots and Systems, 5026/5033 (2012)