Hybrid sliding mode control for a set of interconnected nonlinear components

L. Giacomini, B. Bordbar, D.J. Holding[1]
Aston University, Electronic Engineering,
Aston Triangle,
B4 7ET, United Kingdom
{l.giacomini, b.bordbar, d.j.holding}@aston.ac.uk

Abstract

The paper reports the use of sliding mode control in the design of a hybrid controller for a distributed system with complex and reschedulable task sequences. The approach is demonstrated using a system comprising two loosely-coupled independently-driven mechanisms (or components) which take the form of a pair of coupled inverted pendulums. A sliding mode controller is designed for each pendulum mechanism to provide stabilization and profiled motion control. Then, to accomplish a sequence of tasks, a supervisory system is developed using compositional methods and is modelled and analysed using controlled Petri nets. It is shown that using an appropriate coordination strategy it is possible to achieve a stability envelope for the composite system which is greater than that of the individual components. The function and performance of the system are demonstrated by simulation.

1 Introduction

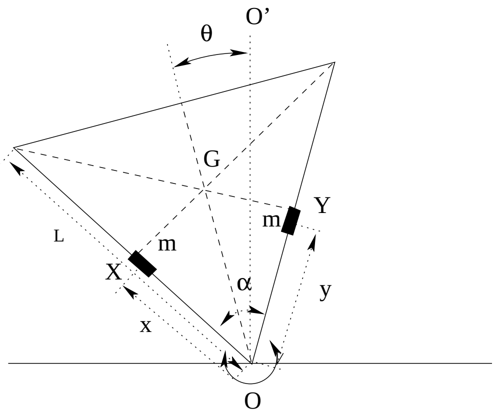
The control of sets of independently driven mechanisms, from a simple conveyor belt to a sophisticated robotic manipulator, is traditionally dealt with in a continuous time/continuous state environment. When these mechanisms are used in a real environment, such as a production line, they have to be co-ordinated and synchronised with eah other and are required to perform sequences of tasks. The control engineer conventionally copes with these requirements by embedding tests (discrete event decision which switch between regulators or reference trajectories) inside the continuous controller. Verification of the functionality of the overall controller is often left to extensive simulation.

In recent years, much research has been carried out on the concepts of *hybrid systems* and *hybrid control* [3], with the objective of developing an integrated approach to the discrete event and continuous parts of such systems. The need is to combine the study of the continuous domain stability/controllability with the study of the intended discrete domain functionality, such as the supervisory control of tasks sequences.

In this paper, we start by analysing the static and dynamic continuous domain behaviour of each of these components of a distributed in isolation and design individual sliding mode controllers for set-point regulation. We then identify conventional switching decisions governing the modes and set-points of the continuous control. Through a process of abstraction we make these decisions an explicit part of the discrete event or supervisory layer, so that they can be modelled and analysed using discrete event methods such as Automata or Petri nets [2, 5, 6].

The paper describes a design study which explores the hybrid control of a distributed system comprising linked inverted pendulums arranged in a production-line style. Section 5.4 describes the development of a supervisory control system for two loosely interconnected pendulum mechanisms and associated product-transfer manipulators using Petri net techniques. The Petri net model of the composite system was then analysed using Petri net techniques to verify the behaviour of the system. Finally, the analysis is extended to examine the effect of severe disturbances, such as those which might cause a pendulum to move into an unstable state (falling or toppling) and which, unless controlled, might cause a 'domino-effect' instability in the second component. It is shown that using an appropriate decision and coordination strategy it is possible to achieve a stability envelope for the composite system which is greater than that of the individual components. Moreover, the reachability graph of the Petri net can be backtracked to identify behaviour (and associated physical parameters) that leads to hazards such as livelock (caused by a deadly mutual embrace of the two inverted pendulums).

[1] Corresponding Author



x distance of mass X from vertex O

y distance of mass Y from vertex O

α angle between the two rails (fixed)

θ angle between OG and OO'

L length of the rails (fixed)

Figure 1: The triangular frame

2 Introduction to the design problem: an inverted pendulum

Consider an inverted pendulum formed by a rotating triangular frame and two balance weights, as shown in Fig. 1. The triangle is assumed isosceles and to lie in the vertical plane. The triangular frame can rotate freely about an axis through the origin O. The mass of the frame is considered to be concentrated in the centre of mass of the triangle. The two balance masses X and Y can move along the sides of the frame: we will call these two sides 'rails'. The movement of X and Y along the rails changes their centre of mass and thus the balance of the triangular frame system. The triangle system is in equilibrium if it is at rest and the effective centre of mass of the frame, X and Y lies on the vertical line OO' (Fig. 1).

Applying the Euler-Lagrange procedure, and taking into consideration some simplyfing assumptions such as the absence of friction and damping terms, the following equations have been derived:

$$\ddot{x} = \frac{F_x}{m} + x\dot{\theta}^2 - gcos\left(\theta - \frac{\alpha}{2}\right) \qquad (1)$$

$$\ddot{y} = \frac{F_y}{m} + y\dot{\theta}^2 - gcos\left(\theta + \frac{\alpha}{2}\right) \qquad (2)$$

$$\begin{aligned} \ddot{\theta} &= \frac{1}{\frac{J}{m} + x^2 + y^2}\Big[- \dot{x}\dot{\theta}x - \dot{y}\dot{\theta}y + gxsin\left(\theta - \frac{\alpha}{2}\right) \\ &+ \quad gysin\left(\theta + \frac{\alpha}{2}\right) + \frac{2ML}{3m}gcos\frac{\alpha}{2}sin\theta\Big] \qquad (3) \end{aligned}$$

where J is the inertia of the frame, F_x and F_y the forces applied to X and Y, respectively. Because of the hard constraint represented by the limited length of the rails, system (1)-(3) is valid only when $0 < x, y < L$. When hitting the boundaries $x = 0, L$, and $y = 0, L$, the quantities $\dot{x}$, and $\dot{y}$ will become 0 instantaneously. When the boundaries are reached, it is natural to withdraw any control force if in the direction of the constrain, however to keep X and Y at the boundaries a force should be applied to counteract the gravity terms, where not already cancelled by the reaction force (i.e. when $x, y = 0$ and the frame is in an upright position, or when $x, y = L$ and the frame is hangig from the pivot).

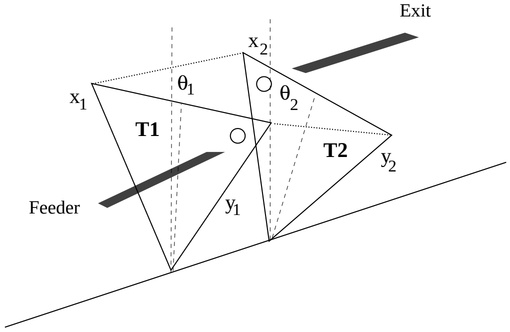


Figure 2:

3 Forming a production-line style composite system

Consider a composite system formed by placing two inverted pendulum (of the triangular form described above) $T1$ and $T2$, on the same axis of rotation but some distance apart, as shown in Fig. 2. Let the vertices of the triangles be linked by non-elastic constraints or chains which have some slack so that the triangles can rotate relative to each other before they become 'locked' at a constant relative displacement. Let the two loosely coupled triangular frame inverted pendulums plus associated product manipulators be configured into a production-line style system as shown in Fig. 2. The task of the system is to move a product from a "feeder" conveyor to an "exit" conveyor via an arbor on each of the triangular frames.

The overall movement that $T1$ and $T2$ have to do to accomplish the target is subdivided into 4 tasks.

[Load_$T1$] **If $T1$ not loaded, $T1$ go to loading point** → the control objective is to move $T1$ to the set point θ_l (*loading point*), and align its arbor with the "feeder" conveyor, using an appropriate motion profile and then for $T1$ to be held stationary at θ_l during the loading period.

[Unload_$T1$] **If $T1$ loaded, $T1$ go to rendez-vous**

point → the control objective is to move $T1$ (using appropriate positional and velocity control) such that it rendez-vous with $T2$ by aligning its arbor with that of $T2$ and then follows it with zero relative velocity for the period of the product transfer from $T1$ to $T2$ (i.e. unload $T1$). Note that, the rendez-vous point is not a fixed position.

[**Load_**$T2$] **If** $T2$ **not loaded,** $T2$ **go to rendez-vous point** → same as **Unload_**$T1$, but $T2$ is exchanged with $T1$. As a result $T2$ is loaded.

[**Unload_**$T2$] **If** $T2$ **loaded,** $T2$ **go to unloading point** → the control objective is to move $T2$ to the set point θ_u (*unloading point*), and align its arbor with the "exit" conveyor, using an appropriate motion profile and then for $T2$ to be held stationary at θ_u during the unloading period.

We shall assume that the dynamics of the product manipulators can be disregarded. For demonstration purposes, the arbors on frames $T1$ and $T2$ are purpose designed to be asymmetric and are arranged such that stability zone for frame $T1$ includes only the load position and for $T2$ includes only the unload position (i.e. $T1$ will topple if moved to the unload position, and $T2$ will topple if moved to the load position). Thus, the production-line task can only be accomplished if both inverted pendulums cooperate.

4 Control Strategy

The control system was designed in stages. First a continuous controller was designed to stabilise a triangular inverted pendulum and enable it to track a particular motion profile. This controller was designed to be suitable for all the specified profiles and was required to satisfy the performance requirement (for example, zero steady state error). The design process involved determining the stability area of the frames and a suitable structure for the controller. To simplify the design problem the original system with independent rails was altered to link the motion of the balance weight on each rail. This had the effect of reducing the original non-holonomic second-order system with two inputs and one output, Eqs. (1)-(3), to a single input single output system. Second, a discrete event or supervisory controller was designed to co-ordinate and synchronise the motion of the pair of linked triangular inverted pendulums and to schedule their motion profiles. The controller was also used to impose a high level control strategy that allowed one triangle to try to rescue the other if it started to fall.

4.1 Preliminaries

The frame will have an unstable equilibrium point at $\theta^* \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ and a stable one at $\pi - \theta^*$. Due to the geometry of the device, θ^* will belong to a finite sector of the plane, that is determinable by moving each of the two masses to the limit on the rails, i.e. $(x, y) = (0, L)$ or $(x, y) = (L, 0)$. Specifically, if the system has initial conditions $(x(0), y(0), \dot{x}(0), \dot{y}(0), \theta(0), \dot{\theta}(0)) = (0, L, 0, 0, \theta_m, 0)$ or, equivalently, $(x(0), y(0), \dot{x}(0), \dot{y}(0), \theta(0), \dot{\theta}(0)) = (L, 0, 0, 0, \theta_M, 0)$, then, under the hypothesis that there are no disturbances, the limiting values for the equilibrium condition are:

$$\theta_M = -\theta_m = arctg\left(tan\frac{\alpha}{2}\left[\frac{1}{1+\frac{2M}{3m}}\right]\right) \tag{4}$$

It follows that the vertical plane can be devided in three sectors

- $\Omega_1 = \{\theta | \ \theta \in [-\pi, -\pi - \theta_m] \bigcup [\pi - \theta_M, \pi]\}$, where all points are stable ones, provided that X and Y are moved to suitable values x, y computable open-loop from a static analysis.
- $\Omega_2 = \{\theta | \ \theta \in (-\pi - \theta_m, \theta_m) \bigcup (\theta_M, \pi - \theta_M)\}$, where the frame surely cannot be stabilized/controlled;
- $\Omega_3 = \{\theta | \ \theta \in [-\theta_m, \theta_M]\}$, the biggest sector in the plane, where the frame can be stabilized/controlled.

4.2 Continuous Time Problem Statement

Given the initial conditions $(\theta_0, \dot{\theta}_0) \in \Omega \times I\!R \subset \Omega_3 \times I\!R$, the control objective is to make the target position $(\theta_d, 0) \in \Omega \times I\!R \subset \Omega_3 \times I\!R$, a stable equilibrium point of the controlled system, where Ω is a suitable subset of Ω_3 to be determined.

5 Dynamic stabilization and motion control

To solve our control problem, a three blocks regulator has been designed as shown in Fig. 3. Each of the rails is controlled with a PD, which provides set-point regulation with a sufficiently fast response time (i.e. *Reg.x* and *Reg.y* for rails X and Y respectively). The set-points are generated by the third block *Reg.θ*. With reference to the scheme in Fig. 3, the three control modules are

$$\begin{array}{ll}
Reg.x & F_x = \begin{cases} -a(x - r_1) - b\dot{x} \ , & 0 \le x \le L \\ gcos\left(\theta - \frac{\alpha}{2}\right) \ , & x = L \\ 0 \ , & x = 0 \end{cases} \\
Reg.y & F_y = \begin{cases} -a(y - r_2) - b\dot{y} \ , & 0 \le y \le L \\ gcos\left(\theta + \frac{\alpha}{2}\right) \ , & y = L \\ 0 \ , & y = 0 \end{cases} \\
Reg.\theta & (r_1, r_2) = f(\theta, \dot{\theta}, \theta^*) \ .
\end{array}$$

where r_1 and r_2 are the reference trajectories (control inputs) for the regulated rails, whilst $\theta^* = [\theta_d \ 0]^T$ are

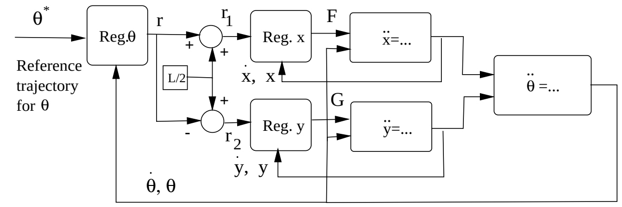


Figure 3: Control scheme

the reference trajectories for the variables θ and $\dot{\theta}$. For example, given $m = M = 1\ kg$, $L = 1\ m$, and $\alpha = \frac{\pi}{2}$, the parameters of the control laws have been chosen as $a = 1000$, $b = 500$.

In choosing an appropriate control strategy for r_1 and r_2, it is of note that if the frame has to balance at an angle at the left of the line OO', the mass Y has to be moved up, and mass X has to be moved down. Alternatively, if θ^* is at the right of OO', mass X should be up, mass Y should be down. Therefore, it is sensible to generally link the two control inputs F_x and F_y (see (1)-(2)) by making $r_1 = \frac{L}{2} + r$ and $r_2 = \frac{L}{2} - r$ where r is a function of $e = \theta - \theta_d$, $\theta_d \in \Omega$, and its first derivative. This reduces the problem to a single input single output one.

5.1 The θ regulator

We want precise set-point regulation, possibly with velocity control as well, and robustness towards bounded disturbances (due to the locking of the two triangles that will be examined in Section 6). For this task, a discontinuous control has been chosen because of the highly non linearity of the system and the fast dynamics it can provide for the closed loop system. (Experiments with a series of PID showed that exact set-point regulation is not possible using PID control).

In developing a discontinuous controller we were conscious of needs of the linked multiple triangular inverted pendulum system. In particular, we intended to examine the co-operative behaviour of the triangles and a scenario (Section 6) in which one triangle rescues the other if should start falling. Therefore, a secondary objective was to make such co-operative behaviour as easy as possible. For example, if one of the two frames is outside the working (stabilisation) area Ω_3, then as the rails cannot be used for stabilisation they may as well be used to minimise the angular velocity of the falling frame. With this in mind, *Reg.θ* can be designed as a two module regulator (the control is switched between them) as follows:

$$r(\theta, \dot{\theta}, \theta^*) = \begin{cases} 0 & \theta \in \Omega_1 \bigcup \Omega_2 \\ u(\theta, \dot{\theta}, \theta^*) & \theta \in \Omega_3 \end{cases}$$

Now, $u(\cdot)$ must be specified.

5.2 Sliding mode control

Given a dynamical system $\dot{x} = f(x) + g(x)u$, $x \in \Re^n$, $u \in \Re$, a typical *sliding mode control* [8] is set up in two steps

1. Design a switching function $S(x)$, such that for $S(x) = 0$ some control objective is satisfied.
2. Design a control law $u(x)$ discontinuous on $S(x) = 0$ such that the states of the system reach the *sliding manifold* $\{x \in \Re^n \mid S(x) = 0\}$ in a finite time.

For step 1, a suitable surface for our system is

$$S(\cdot) = \dot{\theta} + c(\theta - \theta_d)$$

where θ_d is the angle set-point. On the manifold the dynamics of the frame is $\dot{\theta} = -c\theta + c\theta_d$, i.e. an asymptotically stable first order system, then, $\dot{\theta} \to 0$ and $\theta \to \theta_d$.

5.3 Second order sliding mode control

For step 2, because the control u appears only in the second derivative of S (in fact $\dot{S} = \ddot{\theta} + \ldots$, and $\ddot{\theta}$ does not contain u, as can be seen from (3), while $\ddot{S} = \dddot{\theta} + \ldots$ contains it thorough the expressions of $\ddot{x}$ and $\ddot{y}$), we apply a second order sliding mode algorithm [1], that is

1. Let $y_1 = S$ and $y_2 = \dot{S}$.
2. Write the auxiliary system
$$\begin{cases} \dot{y}_1(t) & = y_2(t) \\ \dot{y}_2(t) & = \mathcal{F}[y(t), t] + \mathcal{G}[y(t), t]u(t) \end{cases} \tag{5}$$
with $y(t)^T = [y_1(t)\ \ y_2(t)]$, $y_2(t)$ unmeasurable, $\mathcal{F}[y(t); t] = \frac{\partial}{\partial x}\left(\frac{\partial S}{\partial x}(f(x) + g(x))\right)(f(x) + g(x)u)$, and $\mathcal{G}[y(t); t] = \frac{\partial S}{\partial x} g$ uncertain functions.
3. Find suitable bounds F, Γ_m and Γ_M such that
$$|\mathcal{F}[y(t); t]| < F \tag{6}$$
$$0 < \Gamma_m \le \mathcal{G}[y(t); t] \le \Gamma_M \tag{7}$$
4. Apply the control law
$$\begin{array}{l} u(t) = -\beta(t) U_M \mathrm{sign}\left\{y_1(t) - \frac{1}{2} y_1(t_{M_i})\right\} \\ \alpha(t) = \left\{ \begin{array}{ll} \alpha^* & if\ \ [y_1(t) - \frac{1}{2} y_1(t_{M_i})] \times \\ & [y_1(t_{M_i}) - y_1(t)] > 0 \\ 1 & if\ \ [y_1(t) - \frac{1}{2} y_1(t_{M_i})] \times \\ & [y_1(t_{M_i}) - y_1(t)] \le 0 \end{array} \right. \end{array} \tag{8}$$
where U_M is the control amplitude to be suitably selected, t_{M_i} is such that $y_2(t_{Mi}) = 0$, and $y_1(t_{M_i})$ represents the last extremal value of the $y_1(t)$ function, i.e., the last local maximum, local minimum or horizontal flex point of $y_1(t)$.

In [1] it has been proved that, the corresponding sufficient conditions for the finite time convergence to the sliding manifold are

$$\begin{cases} \beta^* \in (0,1] \cap (0, \frac{3\Gamma_m}{\Gamma_M}) \\ U_M > \max\left(\frac{F}{\beta^*\Gamma_m}, \frac{4F}{3\Gamma_m - \beta^*\Gamma_M}\right) \end{cases} \tag{9}$$

Let us apply this strategy to our case, i.e. let $y_1 = \dot{\theta} + c(\theta - \theta_d)$, and $y_2 = \dot{y}_1$

$$\begin{aligned} \dot{y}_1 &= y_2 && (10) \\ \dot{y}_2 &= \frac{1}{J/m + x^2 + y^2}\big[- ax^2\dot{\theta} - b\dot{x}\dot{\theta} - \dot{x}^2\dot{\theta} - ay^2\dot{\theta} \\ &- b\dot{y}\dot{\theta} - \dot{y}^2\dot{\theta} + g\dot{x}sin(\theta - \frac{\alpha}{2}) + gx\dot{\theta}cos(\theta - \frac{\alpha}{2}) \\ &+ g\dot{y}sin(\theta + \frac{\alpha}{2}) + gy\dot{\theta}cos(\theta + \frac{\alpha}{2}) \\ &+ \frac{2MLg}{3m}\dot{\theta}\; cos\theta\; cos\frac{\alpha}{2}\big] - 3\big[\dot{x}\dot{\theta}x - \dot{y}\dot{\theta}y \\ &+ gxsin(\theta - \frac{\alpha}{2}) + gysin(\theta + \frac{\alpha}{2}) \\ &+ \frac{2MLg}{3m}\; cos\frac{\alpha}{2}\; sin\theta\big]\frac{\dot{x}x + \dot{y}y}{(J/m + x^2 + y^2)^2} \\ &- a\dot{\theta}(x - y)u && (11) \end{aligned}$$

Because of the physical constraints, and of the assumption that we are working in the sector Ω_3, bounds for the expressions in (11) can be easily found. These quantities are functions of $\dot{\theta}(t^*)$, where t^* is the time when a new set-point/reference profile is imposed to the system. As the reader can notice, the control gain is $-\dot{\theta}(x-y)$ i.e. it is a quantity that can be zero and can change its sign, thus it is contrary to the assumptions for the application of the second order sliding mode algorithm describe above. However, this does not prevent the stabilisation of the frame: in fact, the control tends to bring x and y to different values even when the gain is zero. Moreover, the X and Y movement influences directly $\dot{\theta}$, then the condition $\dot{\theta}(x - y) = 0$ will hold for a time of measure zero. As for the sign, because of the geometry of the frame, $\forall\theta \in \Omega_3$, if $x - y > 0$, $\dot{\theta}$ shall become negative, and, if $x - y < 0$, $\dot{\theta}$ shall become positive. Then, apart from transients periods, the applied control is such that $\dot{\theta}(x - y) < 0$, i.e. $u = \beta(t)U_M sign(S - \frac{1}{2}S_{t_{M_i}})$ is applied.

However (11) is the expression of $\ddot{S}$ when the PDs are applied on X and Y. When the ends of the rails are reached,

$$\begin{aligned} \ddot{\theta} &= gxsin(\theta - \frac{\alpha}{2}) + gysin(\theta + \frac{\alpha}{2}) \\ &+ \frac{2MLg}{3m}sin(\theta)cos\frac{\alpha}{2} && (12) \end{aligned}$$

It is easy to see that $\ddot{\theta} = -|\ddot{\theta}|signS$, where $|\ddot{\theta}|$ has a lower bound κ_θ determinable from (12). Then, for each $\theta(t^*)$, we can determine the limit value of $c\dot{\theta}$ for which the term κ_θ is dominating the the expression of $\dot{S}$, i.e. the system satisfies a first order sliding mode condition $S\dot{S} = S(-|\ddot{\theta}|signS + c\dot{\theta}) < -\kappa|S|$, κ some positive constant.

5.4 First order sliding mode control

The reasoning at the end of the previous section, suggests that even a first order sliding mode control law could be applied successfully to the system in question. In fact, let us choose the easiest discontinuous control that is $r = K\ sign(S)$. Because of the constraints and of the fast response of the PD applied to X and Y, $x(= L - y)$, exhibits the behaviour of a saturated function, say σ_ϵ.

$$\sigma_\epsilon = \begin{cases} L, & S \geq \epsilon \\ \kappa\gamma(S) & |S| < \epsilon \\ 0, & S \leq -\epsilon \end{cases} \tag{13}$$

where $\gamma()$ is an increasing function of S, κ is a value proportional to K, and ϵ is proportional to τ/K, τ the PD time constant. After the transients condition (12) holds, i.e. the manifold is attractive, and provided that K is sufficiently large (in other words, the system is allowed to over-saturate as in [7]), $max(\theta) \in \Omega_3$ during the transitory phase, as well as $\dot{\theta}$ below the value $\dot{\theta}_{Max}$ derived from relation (12).

5.5 Coordination and synchronisation logic

The tasks of Section 3 are translated into a series of different set points for the frames. Let θ_1 denote the angular position of $T1$ and with θ_2 the the angular position of $T2$. Then:

$$\begin{aligned} S(\text{Load\_}T1) &= \dot{\theta}_1 + c(\theta_1 - \theta_l) && (14) \\ S(\text{Unload\_}T1) &= \dot{\theta}_1 + c(\theta_1 - \theta_2) && (15) \\ S(\text{Unload\_}T2) &= \dot{\theta}_2 + c(\theta_2 - \theta_u) && (16) \\ S(\text{Load\_}T2) &= \dot{\theta}_2 + c(\theta_2 - \theta_1) && (17) \end{aligned}$$

The system requirement contains also a series of rules that define, in general terms, the logic necessary to coordinate and synchronise the frames and forms the functional requirement of the discrete event part of the hybrid system. Traditionally such logic would have been embedded as switching functions in the continuous controller. However, if the sensor and actuator interfaces are modelled at an abstract level they can be incorporated directly in the discrete event part of the system. To facilitate analysis and reasoning, the interfaces and discrete event system were modelled using Petri nets [6] which have a tangible graphical representation, constructs for describing asynchronous concurrent behaviour, and an underlying mathematical structure.

Using Petri nets the task sequences were modelled and the coordination and synchronisation logic designed as shown in Fig. 4. Transducers are modelled using a

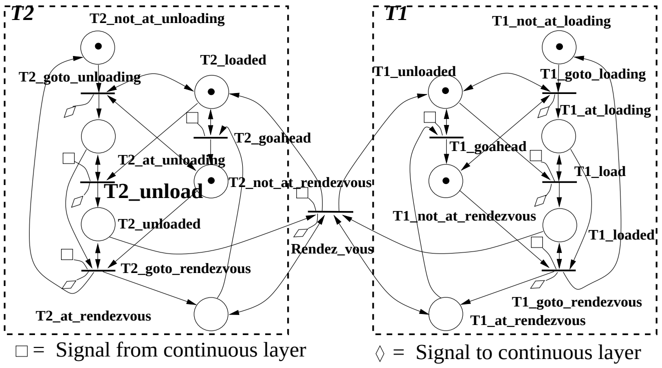


Figure 4:

□ symbol and an arc leading to a transition: when a signal from the continuous layer is received the event "tokenises" the square and thus enables the transition (assuming all input places are tokenised). Actuators or continuous layer controllers are modelled by the symbol ◊ and an arc leading from a transition: when the transition fires ("tokenises" the diamond) it causes a command to be sent to the continuous layer actuator or controller.

5.6 Analysis and verification of synchronisation logic

The Petri net of the task sequence, shown in Fig. 4, consists of two components $T1$ and $T2$, which are synchronised via the transition *Rendez_vous*, which denotes the action of exchanging the component from one triangle to the other. The production–line sequence starts with $T1$ not loaded and located neither at the loading point nor at the Rendez_vous point. Firing of transition *T1_goto_loading* sends $T1$ to the loading point (*T1_at_loading* is marked). On receipt of the sensor signal indicating that $T1$ is at the loading point, *T1_load* fires and issues a command to the continuous layer to say that the component has to be loaded. Next firing of *T1_goto_rendezvous* sends the frame $T1$ to the rendez-vous point and since the triangle is not yet at the loading point, *T1_not_at_loading* is marked. By synchronisation via the *Rendez_vous* transition the component is unloaded from $T1$ and transferred to $T2$ and a new cycle of operation starts for $T1$. For side $T2$ of the Petri net, the behaviour is dual of that of $T1$, as shown in Fig. 4. The Petri net of Fig. 4 is live (free from deadlocks) and safe (has a single, realisable, instance of states) and the reachability graph has 24 states and 41 arcs.

6 Cooperative behaviour in abnormal circumstances: the rescue function

In this section, we introduce the possibility that one of the triangular frames can go past the stability limit angle, perhaps due to an external impulsive disturbance, and start to 'tumble or fall down' (i.e. starts heading towards the stable equilibrium point for the uncontrolled system where it would dangling from the pivot). Because the loosely-coupled triangular inverted pendulums are linked by "chains", it follows that when one triangle begins to fall, the chains would tighten and the triangles would become locked and both would then fall. It follows that the chains provide a mechanism for the "domino" collapse of the system.

However simple appraisal shows that, if the two frames are identical and the ratio between m and M is suitable, the chains provide a mechanism that allows the non-falling triangular frame to come to the rescue of the falling frame. For example, given a method of detecting the onset of "falling" the non-falling triangular pendulum could throw itself in the opposite direction in the hope that the chains might prove to be "safety chains" and rescue the falling pendulum.

At this stage, the following *Falling policy* was introduced,

> If $T1$ ($T2$) is outside the region $[\theta_m, \theta_M]$, (i.e. $T1$ ($T2$) is falling), then $X1$ ($X2$) and $Y1$ ($Y2$) should be moved as fast as possible to $(x, y) = (0, 0)$ (the mechanical constraints imply that $(\dot{x}, \dot{y}) = (0, 0)$)

along with a *Rescue policy*,

> If $T1$ ($T2$) is outside the region $[\theta_m, \theta_M]$, (i.e. $T1$ ($T2$) is falling), then $T2$ ($T1$), should change its target set-point to a predetermined *rescue set-point*; when $T1$ ($T2$) is rescued, $T2$ ($T1$) should go back to try to reach the abandoned target set-point.

6.1 Dynamic model of the two frame system

It is easy to see that the policy introduced for the single frame in Section 5.1, in which X and Y were sent to $(0, 0)$ when outside the stabilizability sector (to minimize the torque of the rails on the falling frame) turns out to be useful for the falling policy. Indeed, the regulator applied to the frames is exactly the same as before. A similar reasoning as the one in section 5.4 can be applied to the compound triangle, to determine a suitable k. This k is used also when the two triangles are not locked. Thus the control law ensures that the dynamics of the single frame is not affected by the disturbance imposed by the other frame.

In Section 2, the biggest stabilization region for one isolated frame was shown to be $[\theta_m, \theta_M]$. A similar static analysis can be applied to the composite two frame system in the case when the frames are locked. This gives

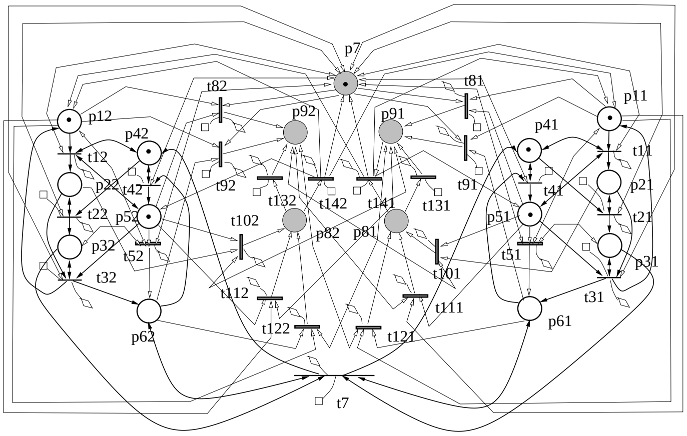


Figure 5:

$M < \frac{3}{2}m\ tan^2\Delta$ as a sufficient condition for the new limit angle

$$\theta_{M_2} = arctg\Big(\frac{tan\frac{\alpha}{2} + tan\Delta\left(1+\frac{2M}{3m}\right)}{1+\frac{2M}{3m} - tan\frac{\alpha}{2}tan\Delta + \frac{2M}{3mcos\Delta}}\Big)$$

to be bigger than the one in (4). Thus, if this condition is satisfied, the composite system of two loosely coupled inverted pendulums have the potential to cooperate and operate over a zone of stability wider than that for the two pendulums operating in isolation.

6.2 Production-line style composite system with Rescue Mode

As a final phase of the design process, the falling/rescue policy was integrated with the task sequence. The resulting Petri net, Fig. 5, has been designed for the discrete part of the hybrid controller. With respect to the Petri Net in Fig. 4, the shaded places have been added along with new transitions. Place $p7$ is marked when none of the frames is in rescue mode. As soon as one among the transitions $t82$, $t92$, $t81$, $t91$ fires because a 'falling down' signal is received from the continuous layer, the token in $p7$ is removed, and no transition in the sub-nets linked to the regular behaviour can fire. Also the tokens in the places $p12$ (*T2_not_at_unloading*), $p11$ (*T1_not_at_loading*), $p22$ (*T2_at_unloading*), $p21$ (*T1_at_loading*), $p52$ (*T2_not_at_rendezvous*), $p51$ (*T1_not_at_rendezvous*), $p62$ (*T2_at_rendezvous*), $p61$ (*T1_at_rendezvous*), are temporarily removed because the condition no longer makes sense during a rescue. Later, when the falling frame is safe, the two subnet will be restarted putting back the tokens in $p7$, $p12$, $p52$, $p11$, $p51$, and tokens are also added the transitions $t52$ and $t51$ to force $T2$ to go to the rendez-vous point if it is unloaded (in the original Petri Net the marking $\{p12, p32, p52, \cdot\}$ gives rise to deadlock), and to force $T1$ to go to the rendez-vous point if it is loaded, respectively.

When one of the two frames is falling down one of $p91$ and $p92$ is marked. Let's say that $T2$ is falling. Then, tokens in $p12$, $p22$, $p52$ are removed from the transitions $t82$ or $t92$, and tokens in the $T1$ sub-net are removed from the only transitions enabled to fire (i.e. one among $t101$, $t111$, $t121$), and place $p81$ is marked, indicating $T1$ in rescue mode. At this stage one of two things can happen: $T2$ comes back to the stabilizability zone (it is said that it has been rescued) and both frames go back to the original tasks (transition $t141$) or $T1$ also goes in the unsafe sector and both frames will fall down (transition $t131$). Analysis of the Petri net of Fig. 5, made with *Design/CPN_3.1.2* (University of Aarhus, Denmark), shows that its reachability graph has 35 states and 103 arcs. It has 4 deadlock configurations, each of which contains the marked places $p91$ and $p92$ which correspond, as expected, to the falling states of the two pendulums.

Analysis shows that the Petri net contains a 'livelock' or 'dynamic deadlock'. This corresponds to the firing sequence $\{t82, t111, t141, t81, t112, t142\}$, in which the two frames are cycling between the falling and rescuing states.

Verification of the system's function also involved searching for any transition firing sequences not containing the transitions $t7$ (*rendez-vous*), which indicates an operational deadlock due to geometric and physical constraints. The Petri net was found to contain such sequences and therefore pre–emption of the sequences was used to ensure correct operation.

7 Implementation with Matlab Toolbox

The continuous system has been implemented as a Simulink (Vers.3) model and comprises models of the two triangular inverted pendulums, the balance weight mechanisms, and the control system. The models of the inverted pendulum have been simplified in the locked case to a composite pendulum model, and particular care has been devoted to the modeling of the lock/unlock conditions that permits switching between the composite model and the two separated models. The control system model consists of a set of simple regulators and lower level mode control switches.

The controlling Petri net has been translated into a Stateflow (Vers.2) diagram. This sends commands (i.e. signals) to the Simulink model (to select between motion profiles and control algorithms). It also receives events (i.e. signals) from the Simulink model. Since Stateflow is not a verifiable tool, the translation from the Petri net to Statechart [4] was achieved by designing a Statechart which implements the Petri net reachability graph.

In Fig. 7, there is the complete statechart with the four

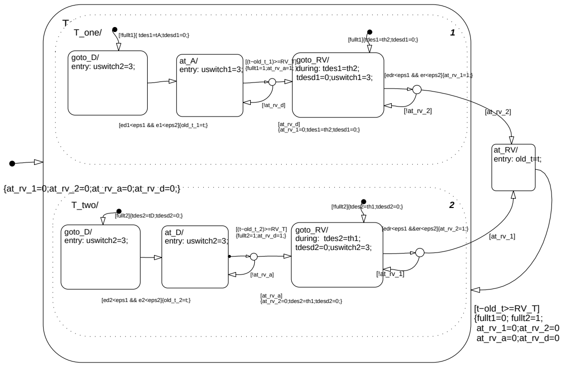


Figure 6:

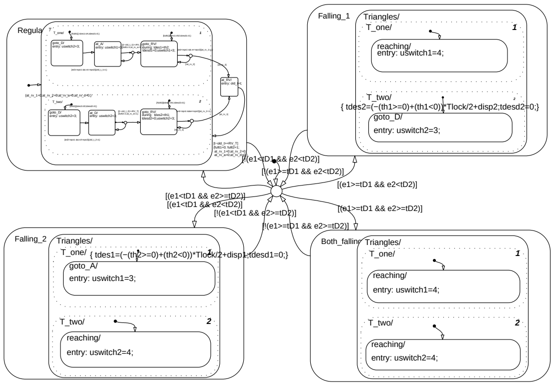


Figure 7:

possible states: *Regular Motion*, *Falling_1* ($T1$ falling down), *Falling_2* ($T2$ falling down), *Both falling*. A test condition is used to switch between the four states at each time step. Going down in the hierarchy, in Fig. 6, there is the enlargment of the part of the chart associated with the regular motion phase. The dotted smooth rectangles indicates two parallel sections. They correspond to the $T1$ and $T2$ controls that run concurrently. Outside the parallel boxes, the state *at_RV* can be seen: it is the synchronisation stage in the *rendez vous.* Extensive testing at limits of working conditions has shown correspondence between the expected behaviour and the implemented one. Figure 8 show the angular position (left) and the angular velocities (right) of the arbors of the two synchronized frames for a regular sequence of movements. In Fig. 9(left) are shown the arbor trajectories for the case that triangle $T2$ starts in an unstable position and the lock angle is not too big (i.e. less than $\frac{\alpha}{2}$). After an initial rescue phase, the falling frame enters the stable area and from that instant on, the two frames follows the regular sequence of set-points. In figure 9(right), $T2$ the lock angle is too small to allow the scheduled task to be completed (in fact the angle between loading point and unloading point is bigger than the lock angle). In this exceptional case, the two frames go to an equilibrium position in which the two arbors are at the minimal distance achievable and the system deadlocks.

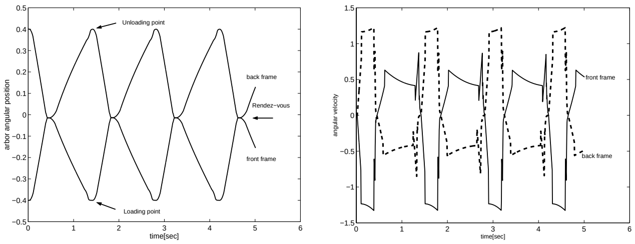


Figure 8: T_1 arbor angle (with respect to the center of the frame)= 0.4, T_2 arbor angle = -0.4, $\theta_1(0) = \dot{\theta}_1(0) = \theta_2(0) = \dot{\theta}_2(0) = 0$, $\alpha_{lock} = \frac{\pi}{2}$, $\alpha = \pi$.

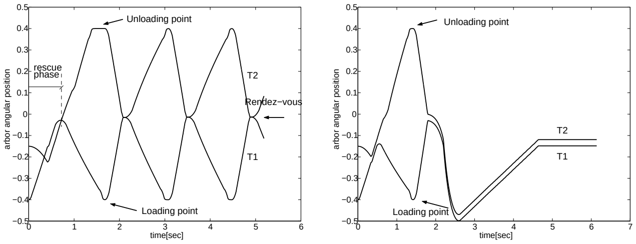


Figure 9: T_1 arbor angle (with respect to the center of the frame)= 0.4, T_2 arbor angle = -0.4, $\theta_1(0) = \dot{\theta}_1(0) = \dot{\theta}_2(0) = 0$, $\theta_2(0) = -0.55$, $\alpha = \pi$, $\alpha_{lock} = \frac{\pi}{2}$ (left) and $\alpha_{lock} = 0.6$ (right).

8 Conclusions

This paper has examined the control of a set of interconnected nonlinear components comprising triangular-form inverted pendulums. It has demonstrated that hybrid sliding mode control can stabilise the inherently unstable pendulums and provide relatively complex supervisory control. The use of Petri nets to make the decision layer accessible to analysis and reasoning is shown to provide invaluable feedback concerning potential behaviour and facilitates the design of a supervisory controller for the composite system. The paper demonstrates how these techniques can be used to design a composite system of linked nonlinear components in which co-operative behaviour increases the stable performance envelope relative to that obtained by the system components acting in isolation.

Acknowledgements The current work has been supported by (UK) EPSRC Grant GR/L31234.

References

[1] Bartolini, G., Ferrara, A., Usai, E.: Applications of a suboptimal discontinuous control algorithm for uncertain second order systems. Int. J. of Robust Nonlin. Control, vol. 7, pp. 299-320 (1997)

[2] Cassandras, C.G.: Discrete Event Systems : Modeling and Performance Analysis. Irwin Publ. (1993)

[3] Grossman, R.L., Nerode, A., Ravn, A.P., Rischel, H.(eds.): Hybrid Systems. Lecture Notes in Computer Science, Vol. 736. Springer–Verlag (1993)

[4] Harel, D.: Statecharts: A Visual Formalism for Complex Systems. Science of Computer Programming, Vol. 8 (1987) 231–274

[5] Koutsoukos, X.D., Antsaklis, P.J.:Hybrid Control Systems Using Timed Petri Nets: Supervisory ControlDesign Based on Invariant Properties. Hybrid Systems V, P. Antsaklis, W. Kohn, M. Lemmon, A. Nerode, S. Sastry Eds.. Lecture Notes in Computer Science, LNCS 1567, Springer-Verlag (1999)

[6] Murata, T.: Petri-Nets: Properties, Analysis and Applications. Proceedings of the IEEE, Vol. 77 (1989) 541–580

[7] De Doná, J.A., Moheimani, S.O.R., Goodwin, G.C., and Feuer, A., "Robust hybrid control incorporating over-saturation," *Sys. and Contr. Letters*,vol. 38, pp. 179-185, 1999.

[8] Utkin, V.I.: Sliding Modes In Control And Optimization. Berlin: Springer Verlag (1992)