

Analisi di automi ibridi  
attraverso l'utilizzo di una libreria dedicata per sistemi cyber-fisici.

Nicola Dessi' VR409182 , Dario Freri VR406704

August 29, 2019

# Introduzione

Questa relazione ha lo scopo di presentare dei sistemi creati con una combinazione di componenti idrici (e non solo) che interagiscono fra di loro in maniera complessa. Lo strumento che viene impiegato é la libreria Ariadne scritta in C++. I sistemi ibridi sono sistemi dinamici nei quali una evoluzione continua é intervallata con eventi discreti innescati da condizioni sugli stati continui. Tipicamente avvengono in applicazioni dove ci sono dei sensori e degli attuatori che interagiscono con il mondo reale. Nel nostro caso abbiamo creato dei sistemi che hanno dei sensori e degli attuatori i quali attraverso delle azioni di controllo ci permettono di avere il comportamento che desideriamo. L'analisi di sistemi ibridi é molto complessa perciò ci avvaliamo di Ariadne come strumento per tale scopo. Infatti questa libreria si basa su rigorosi metodi numerici per lavorare con numeri reali, funzioni ed insiemi nello spazio euclideo, usando tecniche come l'aritmetica sugli intervalli, differenziazione automatica e funzioni di modelli polinomiali come base. Ariadne si basa su tecniche di approssimazione per risolvere problemi di verifica di sistemi ibridi i quali sono non computabili. Data la scarsità di framework *open-source* per risolvere problemi non lineari Ariadne si presenta *open-source* e si basa su tecniche di approssimazione per risolvere problemi di verifica per sistemi ibridi, ed estende le capacità di approssimazione sia alla *upper* che alla *lower semantics* per l'evoluzione di sistemi ibridi. I progetti che vengono presentati in questa relazione sono una composizione di automi-ibridi necessari per creare sistemi complessi i quali vengono supportati da Ariadne, con lo scopo di effettuare un'analisi formale dei suddetti.

## Organizzazione dei sorgenti

Il codice sorgente é suddiviso in moduli:

- `package_name_dir/test.cc` : contiene il *main* e dove vengono settati gli stati iniziali
- `package_name_dir/package_name.hh` : contiene la creazione del sistema ibrido, dove vengono settati i parametri e composti i relativi automi
- `package_name_dir/analysis.hh` : contiene i passi di analisi del sistema
- `package_name_dir/modules` : *directory* contenente i relativi moduli che in genere possono essere 3:
  - `modules/controllers.hh` : contiene la specifica dei controllori
  - `modules/plants.hh` : contiene la specifica degli impianti
  - `modules/valves.hh` : contiene la specifica delle valvole

Ogni oggetto é identificato dal proprio *namespace*: ad esempio per ottenere un controllo di una valvola per l'acqua si invocherà la funzione `water_controller::controller(...)`, mentre per una valvola energetica `energy_controller::controller(...)`.

## Esecuzione

Per eseguire il codice si deve prima compilarlo nell'apposita directory *build*, se non presente deve essere creata. Poi utilizzando `cmake ..` e `make -j` si troveranno dentro la directory *build/bin* i binary dei sistemi creati. Nel dettaglio i sistemi creati sono:

- `crazy_river_hysteresis`: sistema `crazy_river` con controllo a isteresi
- `hydropower_linear_diag`: sistema `hydropower` con città modellata da un sistema lineare con una retta
- `hydropower_linear_exp`: sistema `hydropower` con città modellata da un sistema lineare con evoluzione esponenziale
- `lock_system` : sistema `lock_system`
- `scotland_issue`: sistema `scotland_issue`

## Lock system

*Lock system* rappresenta un sistema di chiuse con l'obiettivo di rendere in grado di far passare una imbarcazione attraverso i diversi livelli di acqua del corso di un fiume. Per formalizzare il sistema sono state fatte delle assunzioni:

- Ogni qualvolta si raggiunge il livello basale una nuova imbarcazione deve essere processata
- Una volta raggiunto il *targetLevel* l'imbarcazione esce immediatamente dalla chiusa

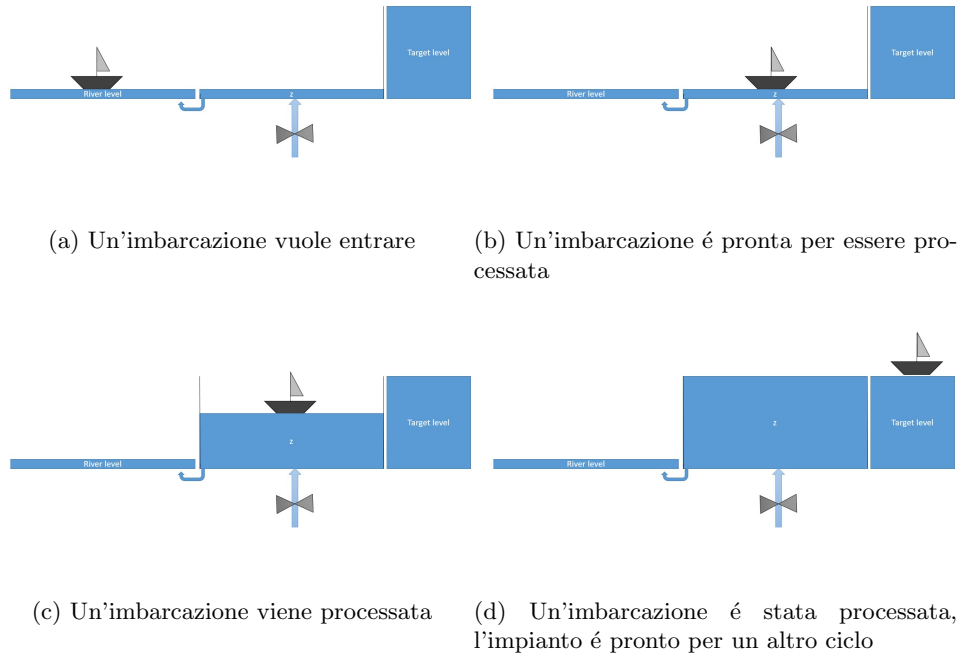
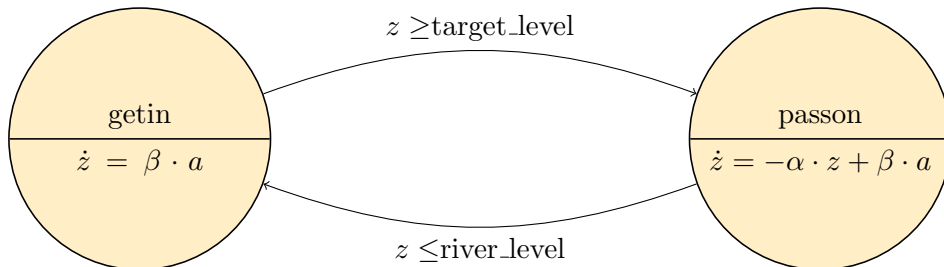


Figure 1: Rappresentazione del sistema

## Impianto

L'astrazione imposta prevede che una tanica rappresenti una chiusa dove l'imbarcazione può immettersi e un sistema idraulico porta il livello dell'acqua al livello *target*. Per la formalizzazione serve una variabile  $z$  che descriva il livello dell'acqua della chiusa; due locazioni per identificare il livello dell'acqua (livello basale, e livello intermedio). Quindi viene definito un ingresso variabile ( $\beta \cdot a$ ), soggetto a controllo; ed una uscita costante utile per far svuotare la chiusa ed essere pronta per una nuova imbarcazione.



## Controllo

Il controllo rappresenta la descrizione del comportamento tipico di una chiusa. Ovverosia al passaggio della barca abbiamo un controllo proporzionale in grado di aumentare il livello dell'acqua a quello desiderato (*targetLevel*); ed uno stato *idle* dove il controllo si ferma per permettere lo svuotamento della chiusa e quindi l'ingresso della imbarcazione successiva.

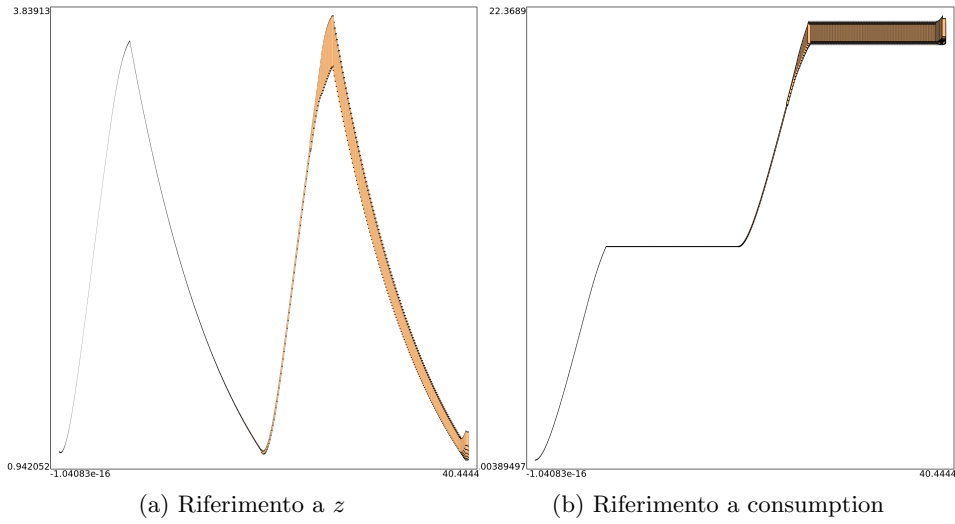
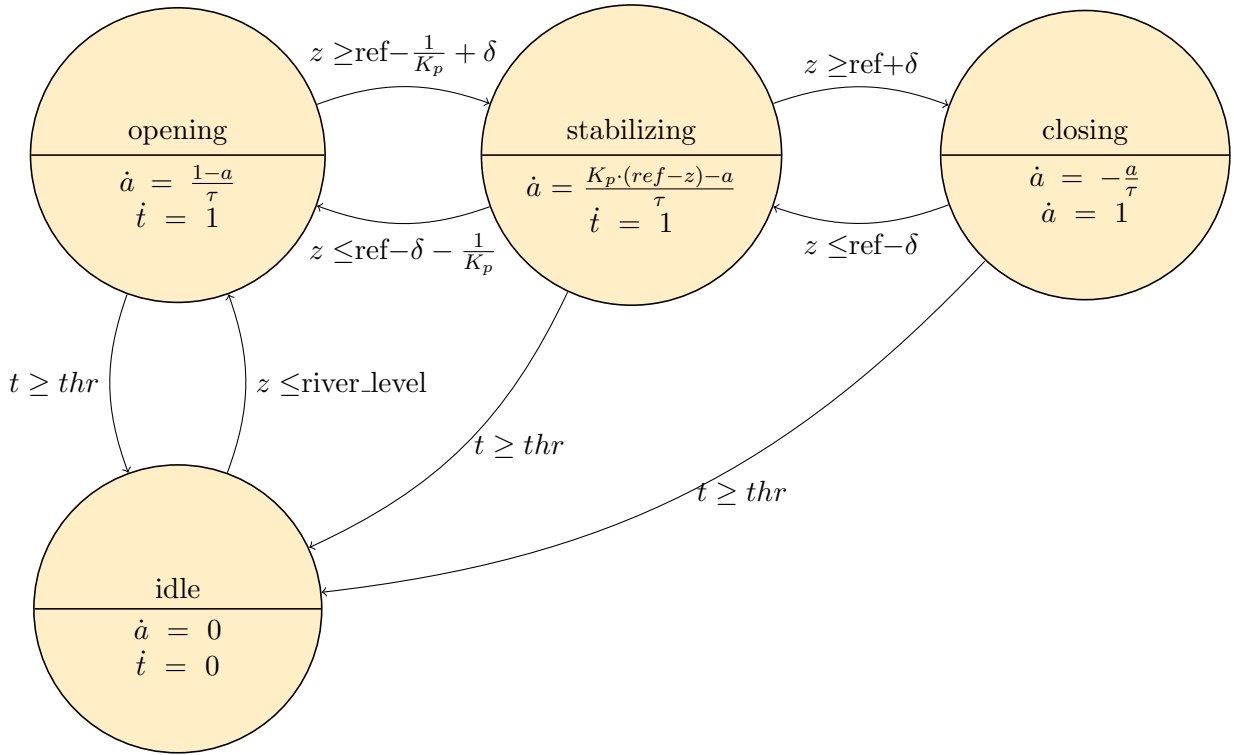


Figure 2: Evoluzione di 30 secondi del sistema



## Analisi

Lo scopo dell'analisi é lo studio del consumo energetico legato al funzionamento della valvola. In particolare si vuole vedere nel tempo il consumo energetico della pompa idraulica al variare dell'apertura e chiusura della valvola. Viene assunto che il consumo energetico sia costante ed in particolare il calcolo é eseguito in questo modo:  $a \cdot 2$  dove  $a \in [0, 1]$  é l'apertura della valvola ovvero della pompa idraulica e 2 é la potenza richiesta (banalmente negli esempi é di 2KWh). Nelle figure [2a] e [2b] viene mostrata l'evoluzione del sistema per 30 secondi:

Si nota chiaramente che il comportamento é quello atteso e rispettivamente ([2a]) la tanica si riempie per far salire la barca e si svuota ciclicamente; mentre il consumo energetico accumulato accresce o resta fermo a seconda della necessità.

**Analisi passo d'integrazione:** Analizzando l'evoluzione del sistema si é notato che il raggio aumentava rapidamente e questo portava a transizione non ben definite; la soluzione é stata dare piú accuratezza numerica decrementando il passo di integrazione. In figura [3a] si mostra l'evoluzione del consumo energetico e della sua derivata; sebbene il sistema con questo settaggio evolva, si nota un oscillazione (figura [3c]) dovuta

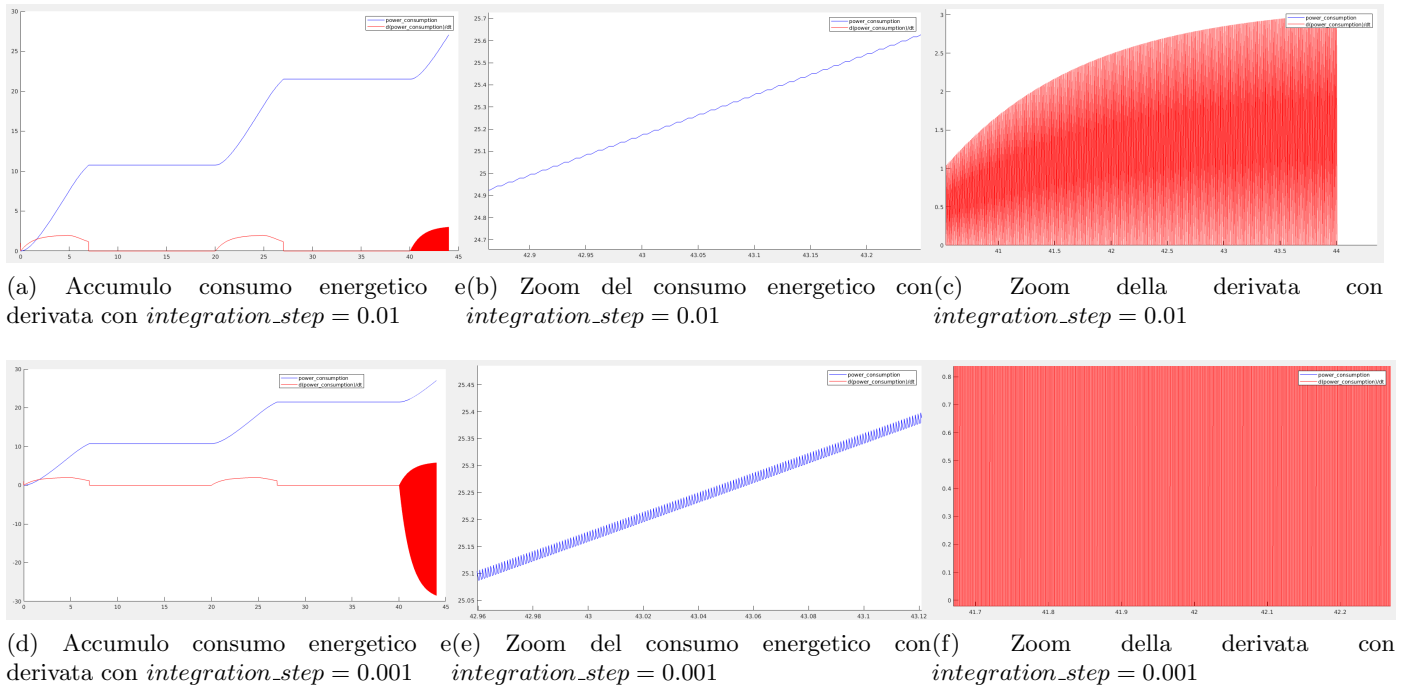


Figure 3: Evoluzione di 40 secondi del sistema

al fatto che il controllore nella fase di stabilizzazione apre e chiude la valvola (figura [3b]) per mantenere il sistema nel livello di riferimento, questo si traduce in un'oscillazione che complica i calcoli nella fase di risoluzione integrale del sistema nel calcolo dell'orbita. Diminuire ancora di più il passo di integrazione non si è rivelata una soluzione, infatti stabilizzando il sistema l'oscillazione risulta maggiore come mostrato nelle figure [3d] [3e] [3f]

## Relazioni tra le variabili

( $a, z$ ), **figura [4]**: dallo stato iniziale (cerchiato in rosso), che identifica la presenza dell'imbarcazione nella tanica, la valvola si apre permettendo l'aumento del livello fino al raggiungimento della soglia *target*. Una volta raggiunta tale soglia scatta l'evento che porta il controllore allo stato *idle*, dove viene resettata la variabile  $a$ , e quindi si nota un salto verso l'asse delle ordinate e gli stati raggiungibili riprendono da lì per tornare verso lo stato iniziale e quindi riprendere il ciclo (processare un'altra imbarcazione).

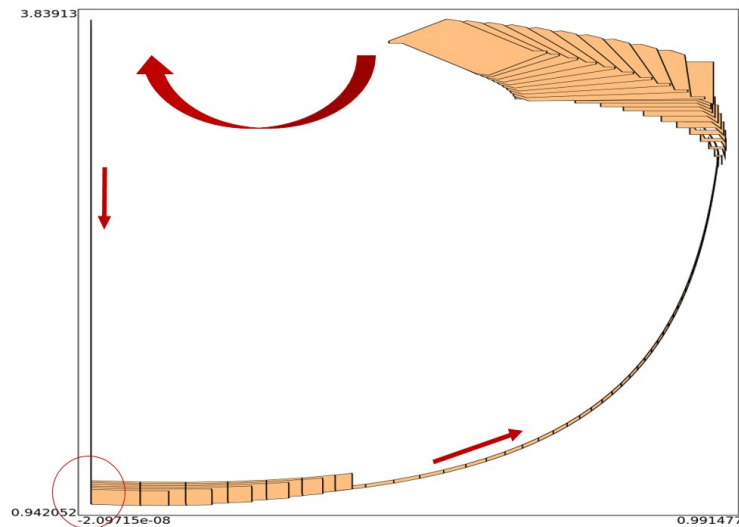


Figure 4: Relazione tra  $a$  e  $z$

$(consumption, z)$ , **figura [5]** : partendo dallo stato iniziale per aumentare il livello dell'acqua viene richiesta energia per cui il consumo energetico e il livello dell'acqua aumentano all'unisono fino al raggiungimento della soglia, dopo il quale il consumo energetico resta fisso, mentre il livello dell'acqua decresce in maniera costante. Questo comportamento é ripetitivo.

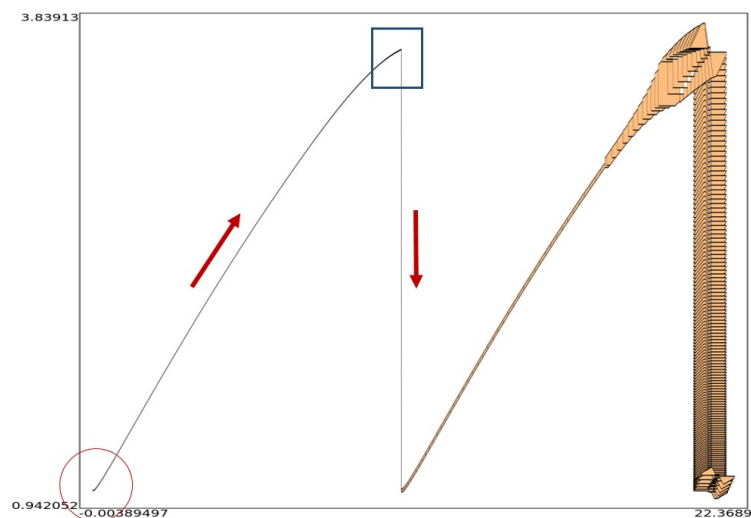


Figure 5: Relazione tra  $consumption$ ,  $z$

# Crazy river

Crazy river modella un sistema di piscine collegate da scivoli. Per questo studio sono state utilizzate due piscine a cascata; tutte le piscine hanno un'uscita costante verso una terza tanica di supporto responsabile del filtraggio dell'acqua che verrà poi riutilizzata come ingresso per la prima piscina. Il controllo del sistema ingegneristico di filtraggio serve per garantire l'utilizzo a fini pubblici o commerciali. In figura [6] viene data una rappresentazione grafica.

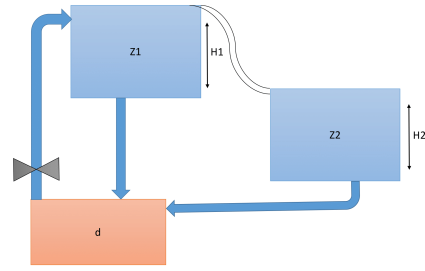
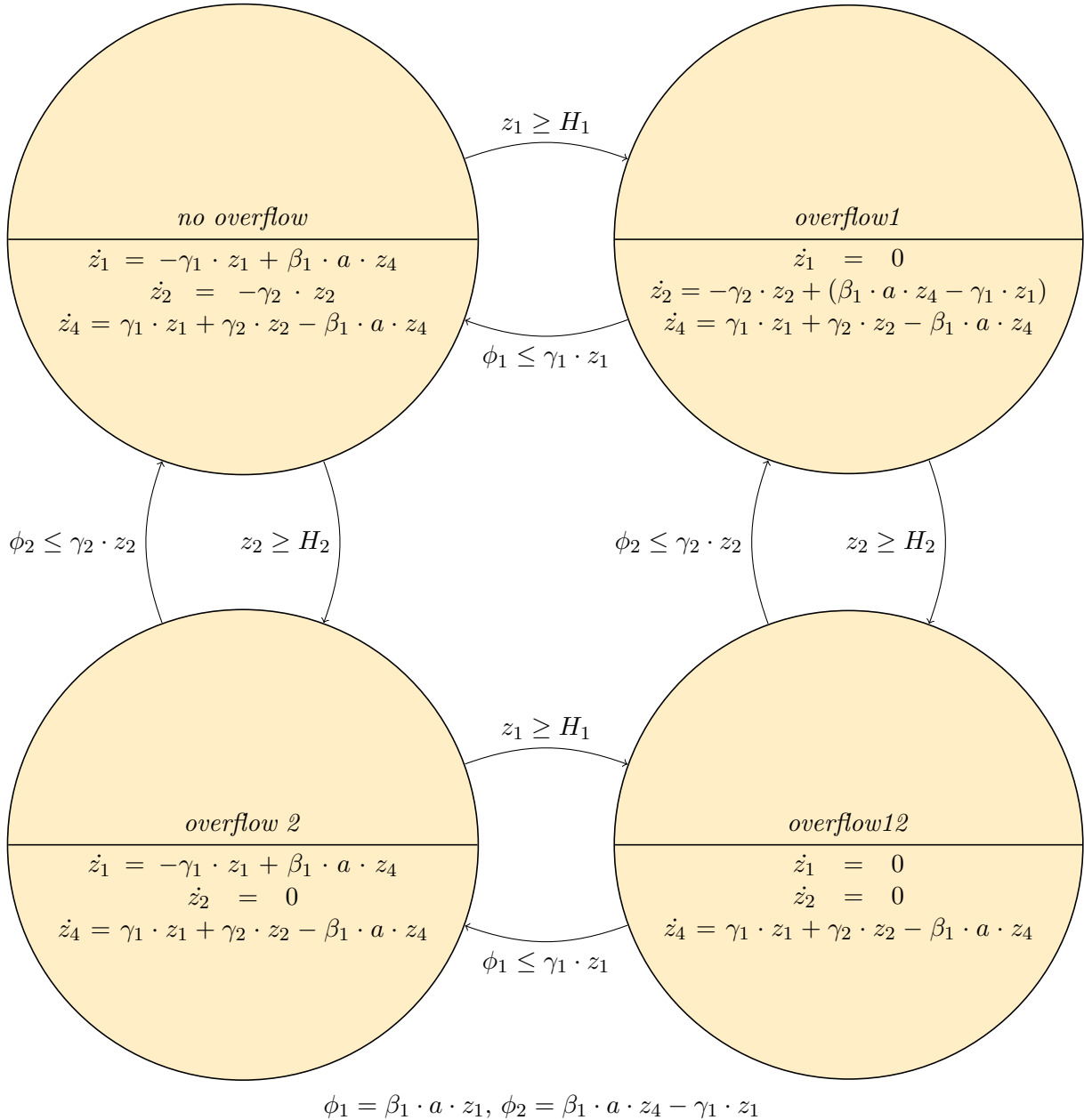


Figure 6: Rappresentazione grafica

## Impianto

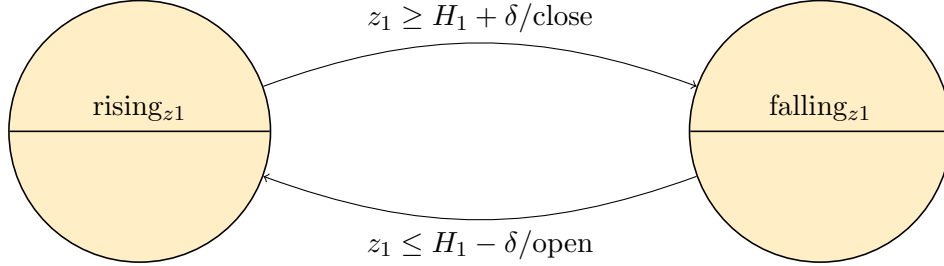
La prima piscina ( $z_1$ ) ha un output costante verso la tanica di supporto ( $z_4$ ) e ha un ingresso modulato da un controllo, inoltre se raggiunge il livello di overflow scarica sulla seconda piscina attraverso lo scivolo. La seconda piscina invece ha un'unica uscita costante verso la tanica di supporto e un ingresso dipendente dall'overflow della prima piscina. La situazione che si vuole evitare é che la seconda tanica vada in overflow, poiché dell'acqua andrebbe persa e la tanica di supporto si troverebbe senza acqua sufficiente per l'utilizzo.



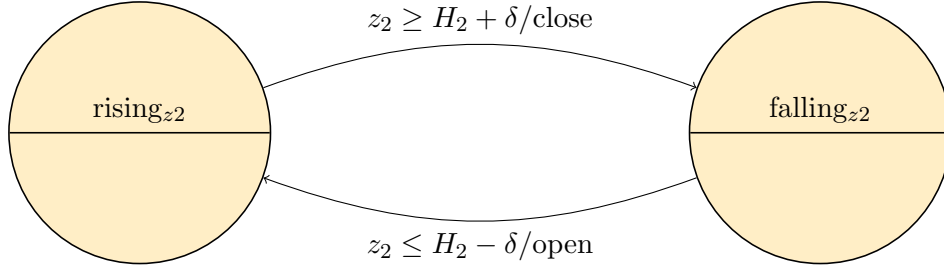
## Controllo

Il controllo mira ad ottenere un livello minimo ed un livello massimo per entrambe le piscine e quindi si é preferito utilizzare un controllo a isteresi. Per tale sistema sono necessari 3 controlli, uno per ogni tanica. In tale contesto risulta molto utile la funzione *compose* fornita da *Ariadne* in quanto si possono modellare separatamente tali controlli e quindi comporli direttamente senza realizzare a mano il controllo cosí composto.

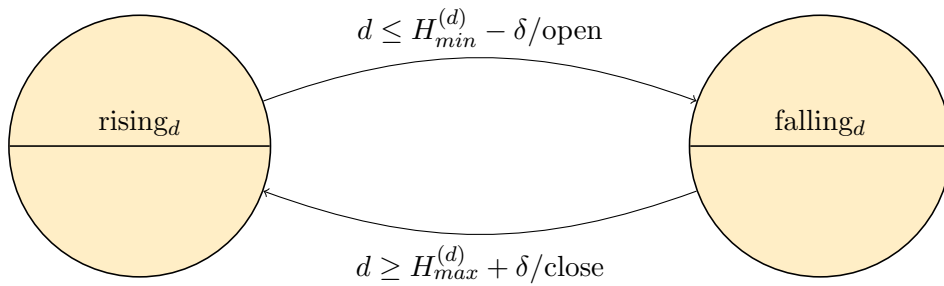
**Controllo per  $z_1$ :** Il controllo per la prima piscina impone un livello minimo ed un livello massimo oltre il quale il controllo agisce riportando il valore dentro il range accettabile:



**Controllo per  $z_2$ :** Il controllo per la seconda piscina é analogo alla prima piscina, quindi:



**Controllo per  $d$ :** Il controllo per il depuratore é leggermente diverso: analogamente alle altre taniche si vuole garantire un intervallo di valori accettabili entro il quale il controllore deve garantire la presenza, tuttavia aprendo la valvola il livello diminuisce e quindi il controllo risulta inverso rispetto a quelli visti precedentemente:



## Analisi

Lo scopo del sistema Crazy River é quello di mantenere un livello accettabile nei due tank che rappresentano le piscine per mantenere un livello di sicurezza accettabile per un uso pubblico dell'acqua park. Inoltre sará presente un clorinatore che effettuerá il filtraggio; ed anche in questo caso avrá bisogno di un livello minimo di acqua per non far bruciare i motori. Il controllo deve minimizzare la perdita d'acqua causata dall'overflow della seconda tanica. Il settaggio del sistema per le successive analisi é il seguente:

- $initial\ box^1 = \{a = [0.0, 0.0], d = [1.75, 1.75], z_1 = [1.75, 1.75], z_2 = [1.5, 1.5]\}$
- Parametri:  $\{\beta_1 = 0.4, \gamma_1 = 0.04, \gamma_2 = 0.04, h_{min} = 1.0, h_{max} = 2.0, \delta = 0.1, H_1 = 1.9, H_2 = 1.9, H_{min}^{(d)} = 1.0, H_{max}^{(d)} = 2.0\}$

---

<sup>1</sup>Insieme di stati iniziali



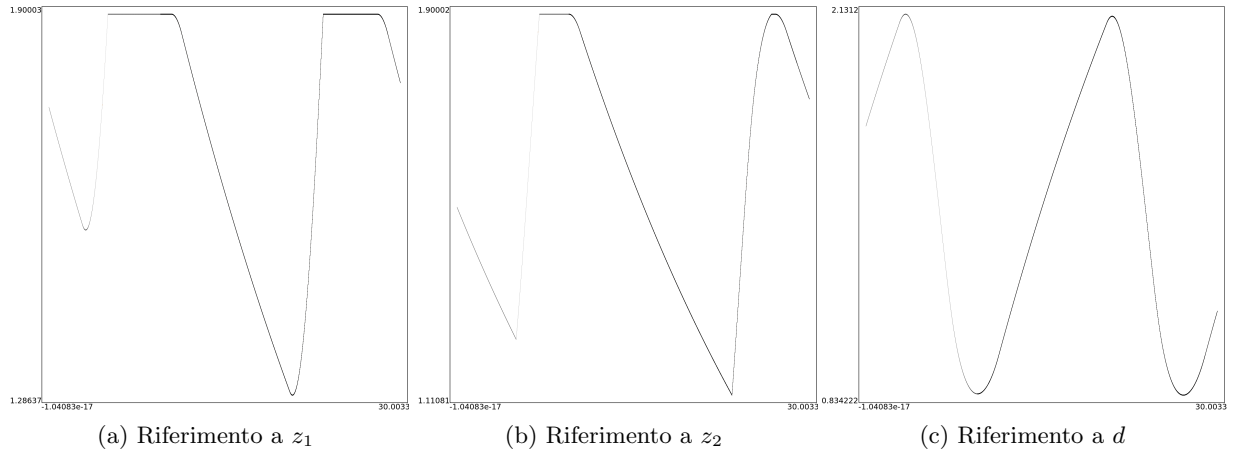


Figure 7: Evoluzione di 30 secondi del sistema

## Intervalli

In riferimento alla prima tanica ( $z_1$ ) nell'evoluzione degli stati raggiungibili [figura: 7a] si nota subito che l'intervallo accettabile é rispettato.

In riferimento alla seconda tanica ( $z_2$ ) nell'evoluzione degli stati raggiungibili [figura: 7b] si nota che per alzare il livello dell'acqua si sfrutta l'overflow della prima tanica, inoltre il periodo in cui il sistema resta in overflow tende a diminuire, secondo le specifiche del controllo. In entrambi i casi ( $z_1$ ,  $z_2$ ) l'andamento é periodico, come desiderato. In riferimento al clorinatore ( $d$ ) l'evoluzione degli stati presenta un andamento periodico (*sinusoidale*) [figura: 7c], motivato dal fatto che il flusso in uscita poi rientra nel clorinatore stesso nel momento in cui la valvola viene chiusa.

## Relazioni tra $a$ e $\{d, z_1, z_2\}$

La figura [8] rappresenta l'andamento ripetitivo del sistema relazionando  $a$  con  $z_1$ . Partendo dagli stati iniziali (cerchiati in rosso) all'aumentare di  $a$  si nota un incremento significativo di  $z_1$  fino al raggiungimento della soglia di *overflow*, dopo la quale  $a$  non influenza più  $z_1$ . Invece decrementando  $a$  si ha uno *shifting comportamentale* del sistema che delinea lo svuotamento della prima tanica. É da notare che il fattore decrementale di  $z_1$  é  $\gamma_1$ , il quale essendo molto piccolo (10 volte più piccolo del flusso in ingresso) rende lo svuotamento molto lento.

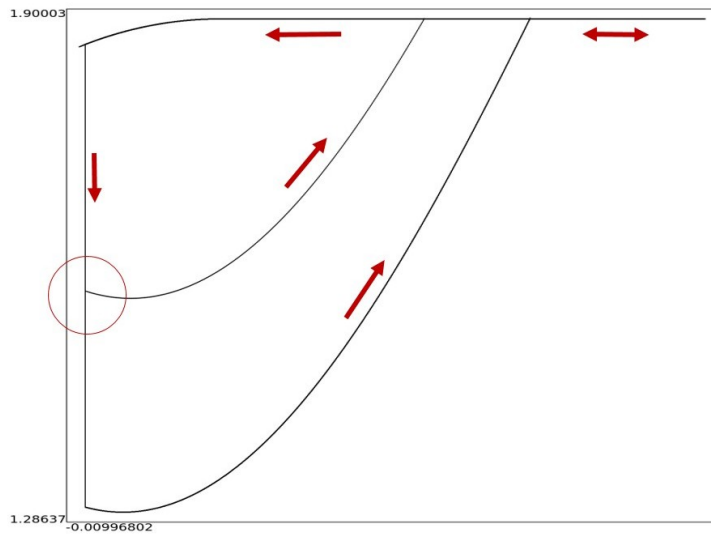


Figure 8: Evoluzione di 30 secondi del sistema relazionando  $a$  con  $z_1$

La figura [9] rappresenta la relazione che sussiste tra  $z_1$  e  $z_2$ ; non verrà presa in considerazione la relazione ( $a$ ,  $z_2$ ) poiché essa dipende indirettamente, essendo  $z_2$  dipendente da  $z_1$  e  $z_1$  dipende da  $a$ . Partendo dagli stati iniziali (cerchiati in rosso) si nota chiaramente che solo quando  $z_1$  va in overflow (lato destro della

figura) la seconda tanica si riempie e non appena  $z_1$  passa sotto la soglia,  $z_2$  inizia a decrescere finché  $z_1$  non ritorna in *overflow*.

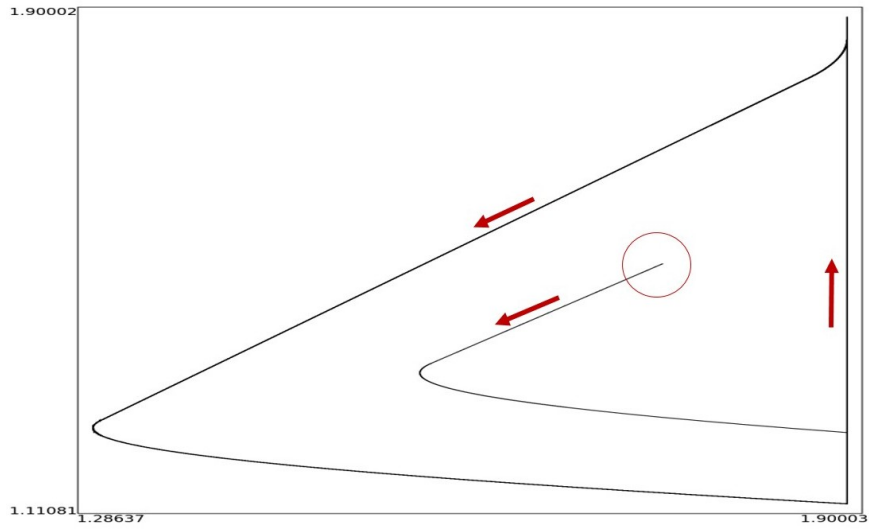


Figure 9: Evoluzione di 30 secondi del sistema relazionando  $z_1$  con  $z_2$

La figura [10] rappresenta la relazione tra  $a$  e  $d$ , e si nota un andamento ciclico (lo stato iniziale é  $(0,0)$  ). Viene sottolineato il punto di minimo (linea tratteggiata in blu), il quale delinea i due semipiani dove nel lato destro il flusso in ingresso di  $d$  é minore del flusso in uscita; nel semipiano sinistro il contrario.

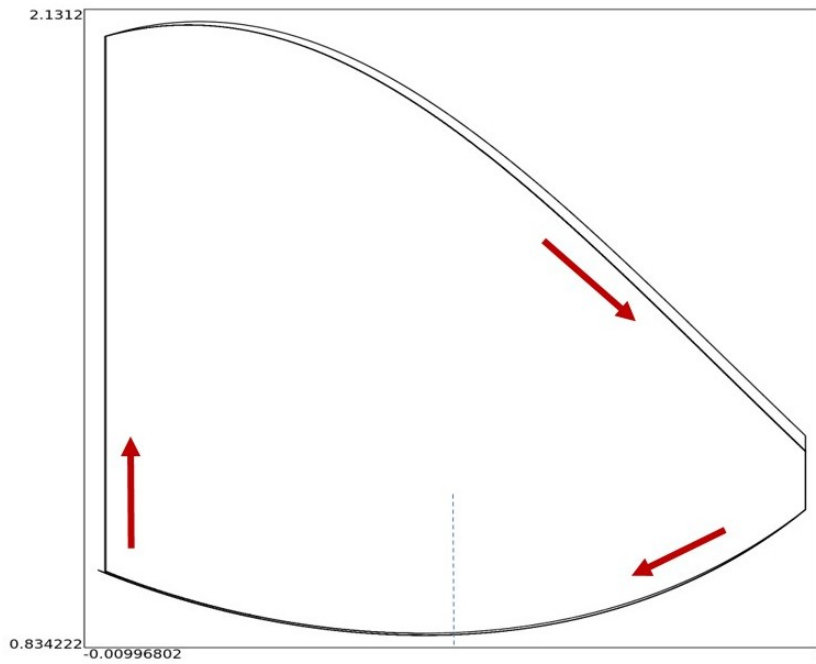


Figure 10: Evoluzione di 30 secondi del sistema relazionando  $a$  con  $d$

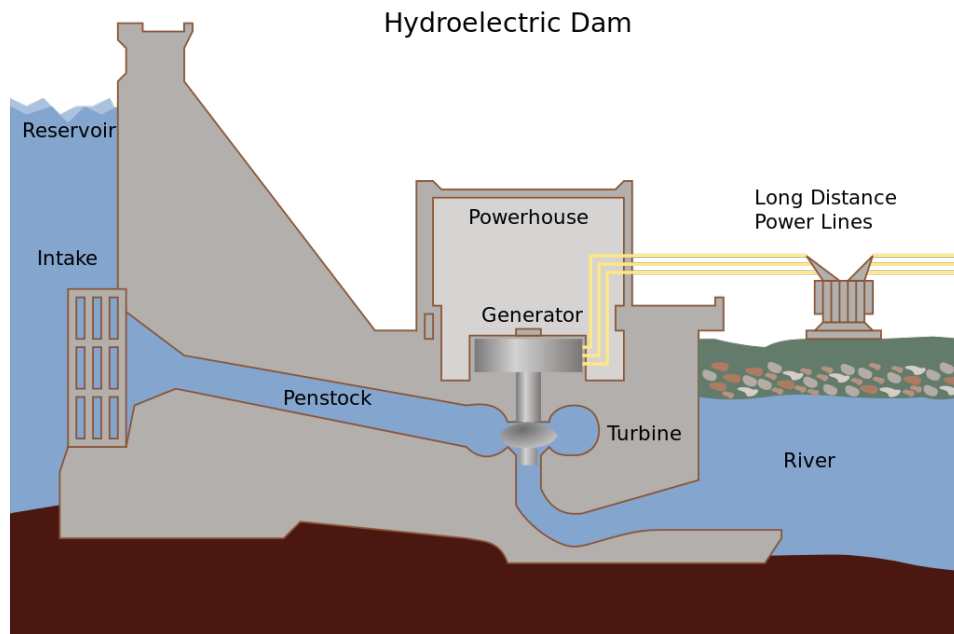
# Hydropower

Il sistema *hydropower* mira a modellare la produzione energetica posta a valle di una diga correlando il flusso di acqua uscente dalla diga con la potenza prodotta secondo la nota formula :  $P = \eta \cdot \rho \cdot Q \cdot g \cdot h$  dove:

- $P$ : potenza generata
- $\eta$ : é l'efficienza della turbina (  $\eta \in [0, 1]$  )
- $\rho$ : é la densità dell'acqua in kilogrammi per metro cubo
- $g$ : é l'accelerazione gravitazionale
- $h$ : é l'altezza da cui l'acqua cade per cadere in caduta libera sulla turbina

L'idea é astrarre un accumulatore di energia e immaginarlo come un *tank* il cui flusso in ingresso é la potenza generata dalla turbina. Il sistema deve garantire un livello minimo di energia accumulata e un livello massimo dopo il quale non é piú sicuro accumulare energia. L'energia accumulata viene impiegata da una città modellata da un automa.

Figure 11: Sistema di un impianto idroelettrico



## Specifica

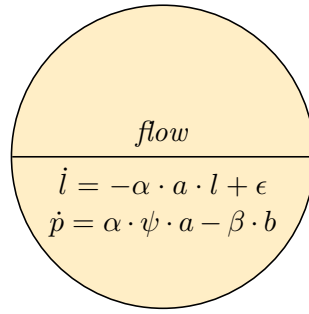
Verranno analizzati gli automi che descrivono il sistema sopra descritto:

### Diga, turbina e accumulatore

Astraendo il problema si intuisce subito che modellare la turbina non é necessario in quanto la formula presentata sopra fornisce già una costante in grado di correlare il flusso uscente dal lago e la potenza entrante nell'accumulatore; quindi le variabili in gioco sono soltanto 2:

- $l$ : la quantità di acqua presente nel lago
- $p$ : l'energia accumulata fornendo acqua o meno alla turbina

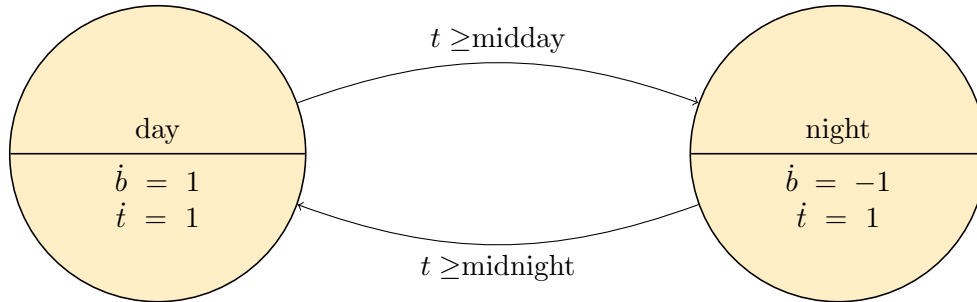
Quindi entrando nel dettaglio il lago presenta un'uscita variabile modulata da una valvola ed un ingresso costante (i vari fiumi affluenti al lago); l'accumulatore ha il flusso in ingresso descritto correlando il flusso in uscita del lago con la potenza prodotta e una variabile  $b$  che descrive il consumo energetico richiesto da una città.



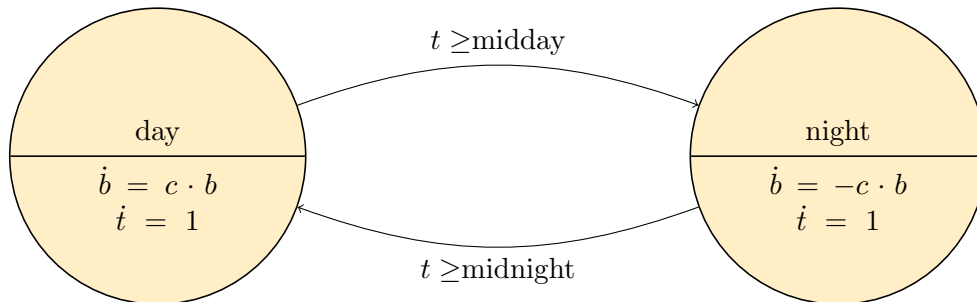
## Descrizione della città

La descrizione della città é molto complicata, tuttavia si può immaginare un andamento periodico. Inoltre é necessario astrarre il modello per semplificare il calcolo della variabile in considerazione. Quindi di seguito verranno presentate diverse astrazioni che simulano un andamento periodico. In tutti i modelli successivi si ipotizza che l'utilizzo energetico sia monotono crescente nell'intervallo  $[0, \frac{T}{2}]$  e monotono decrescente nell'intervallo  $(\frac{T}{2}, T]$ , avendo quindi un unico punto di massimo nell'intervallo  $[0, T]$  che é il picco giornaliero raggiunto a mezzo giorno (o mezzanotte).

**Lineare:** Viene modellato un andamento a *dente di sega* del tipo  $b(t) = t$  per la parte crescente e  $b(t) = -t$  per la parte decrescente.

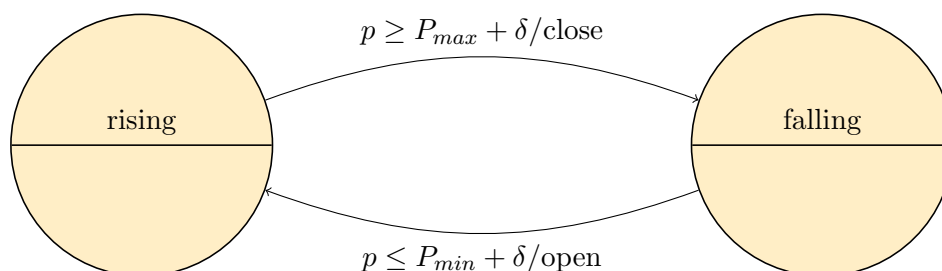


**Evoluzione esponenziale (sistema lineare):** Viene modellato un andamento del tipo  $b(t) = e^{c \cdot t}$  per la parte crescente e  $b(t) = e^{-c \cdot t}$  per la parte decrescente.



## Il controllore

Il controllore esegue un controllo *hysteresis* dove cerca di mantenere i livelli energetici in un determinato range  $[P_{min}, P_{max}]$  modulando la valvola che svuota il lago nella turbina.



## Analisi

Come mostrato nell'evoluzione del sistema fino a 300 secondi i limiti imposti sono rispettati. Per l'evoluzione sono state utilizzate le impostazioni di default:

- **parametri:**  $\{g = 9.81, \eta = 0.85, h = 10, \rho = 1000.0, \alpha = 0.1, \beta = 1.0, \epsilon = 0.2, \gamma = 0.01, P_{min} = 30000, P_{max} = 100000\}$
- **Initial set box:**  $\{a = [0.0, 0.0], b = [0.1, 0.1], l = [100.0, 100.0], p = [50000.0, 50000.0], t = [0.0, 0.0]\}$

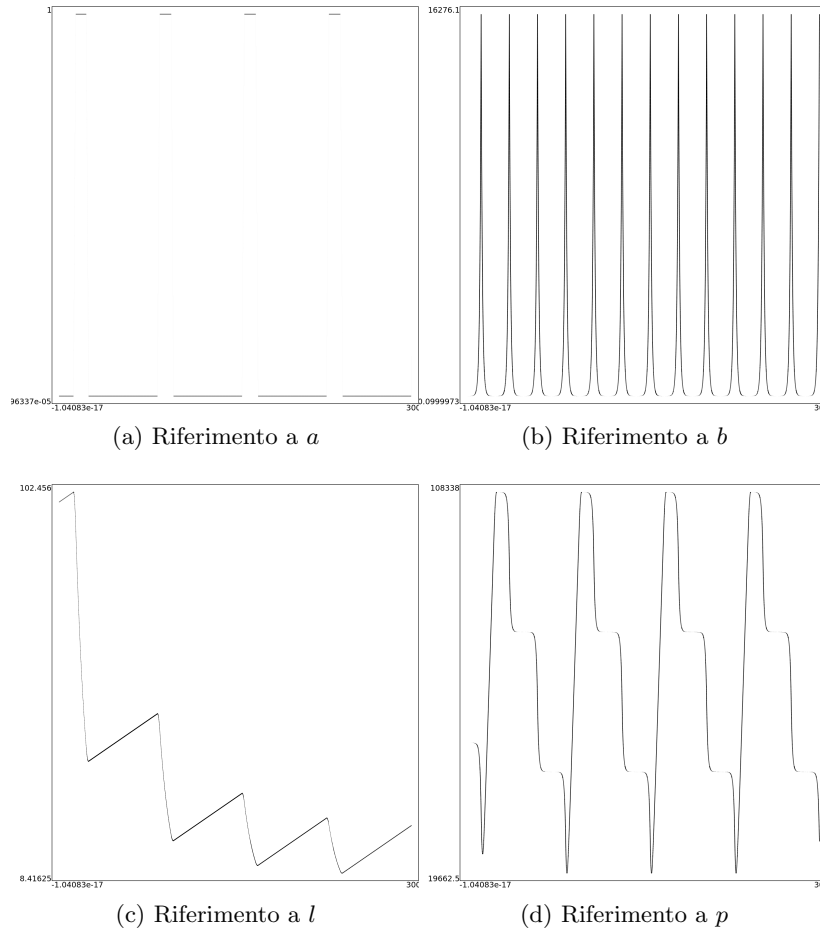


Figure 12: Evoluzione di 300 secondi del sistema

## Esponenziale

$(a, b)$  : nella figura [13] viene presentata relazione che sussiste tra  $a$  e  $b$  é periodica e partendo dagli stati iniziali indicati dal cerchio rosso l'automa che simula la città inizia a crescere di valore molto rapidamente lungo l'asse delle ordinate fino ad un picco di 16276.1, dopo il quale (a mezzo giorno/mezza notte) inizia a decrescere. Una volta raggiunto il livello minimo accettabile di energia accumulata il controllo agisce sulla variabile  $a$  (asse delle ascisse) aumentandone il valore, e tale evento é indicato dal *box* blu. Successivamente la valvola viene chiusa mentre l'automa della *città* inizia salire fino al valore massimo dopo di che inizia a scendere; una volta raggiunto l'evento dove la soglia minima é superata si ripete il ciclo di prima.

$(a, p)$  : nella figura [14] viene delineata la relazione tra  $a$  e  $p$  é ciclica e partendo dallo stato iniziale indicato con un cerchio rosso il controllo apre e chiude la valvola mantenendo il sistema in equilibrio dentro i limiti della *box*.

$(a, t)$  : nella figura [15] la relazione mostra l'attività della valvola scandita in una giornata, e come previsto é maggiore dopo *mezzo giorno*.

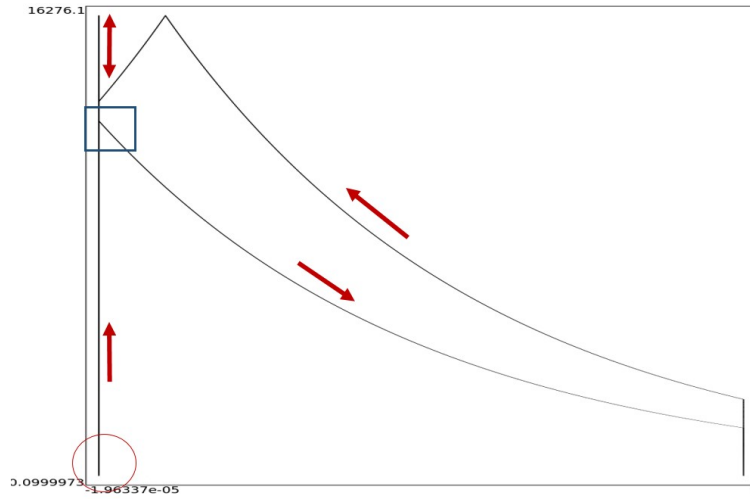


Figure 13: Relazione tra  $a$  e  $b$

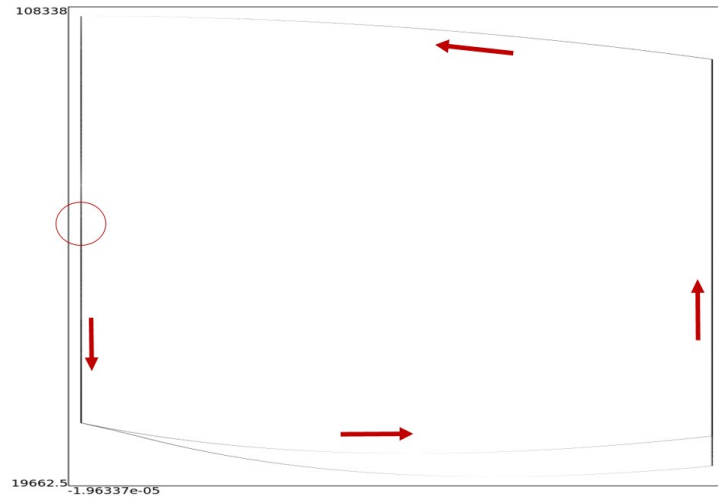


Figure 14: Relazione tra  $a$  e  $p$

$(b, p)$  : nella figura [16] partendo dallo stato iniziale indicato dal cerchio rosso, si nota la linearità del consumo energetico: seguendo le linee rosse la quantità di energia decresce linearmente fino ad un livello minimo dove aprendo la valvola il livello viene portato al massimo lungo la retta che percorre l'asse delle ascisse.

$(t, b)$  : la figura [17] rappresenta la caratteristica dell'automa che modella la città, infatti rispetto a quanto detto nella specifica del sistema questo grafico rappresenta il segnale periodico nell'intervallo  $[0, T]$  con picco massimo in  $\frac{T}{2}$ .

### Rampa

L'analisi della *rampa* è molto simile alla precedente, di conseguenza non verrà analizzata in quanto cambia solo la caratteristica dell'automa che simula la città, e poiché l'evoluzione già analizzata è più complessa verrà presentata solo quella.

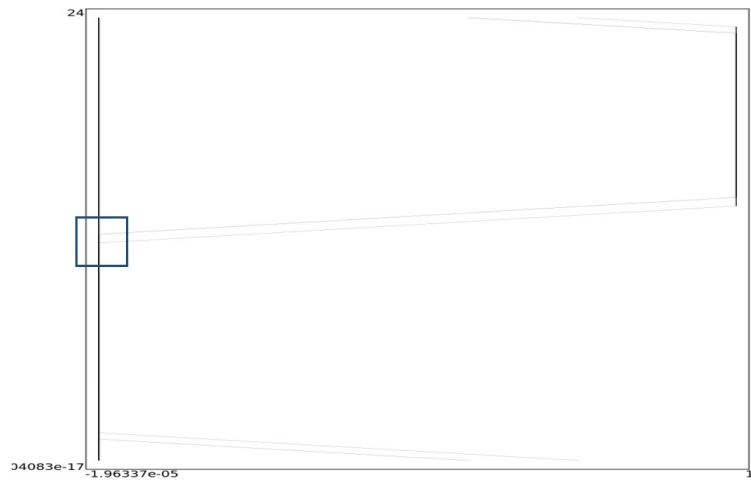


Figure 15: Relazione tra  $a$  e  $t$

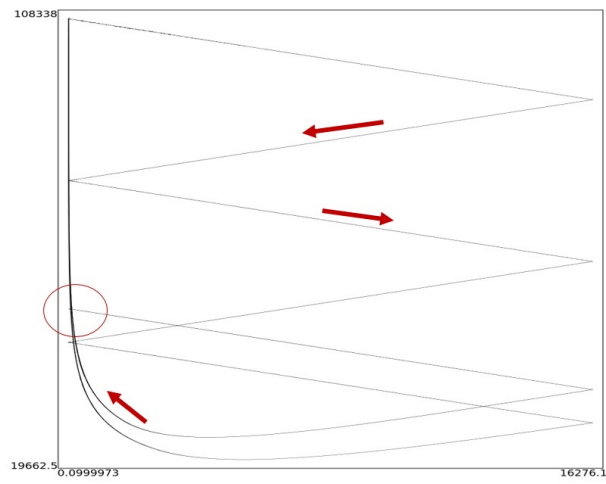


Figure 16: Relazione tra  $b$  e  $p$

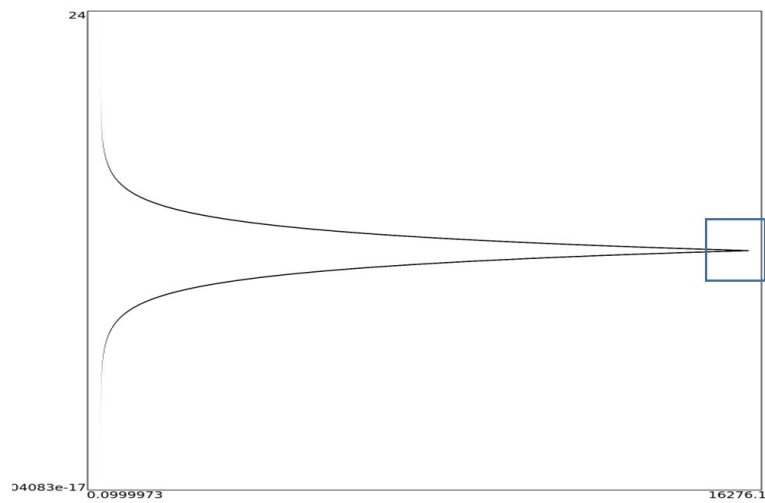


Figure 17: Relazione tra  $t$  e  $b$

## Scotland issue

Il sistema scotland's issue modella una piscina all'aperto, dove é presente la possibilità di pioggia. L'obiettivo é di modellare ed analizzare un controllore il quale cerca di garantire il minimo spreco possibile ed un livello minimo di acqua. La perdita di acqua é considerata spreco ed é causata da un'*overflow*. Nel sistema é presente un'uscita costante verso un ipotetico *clorinatore* rappresentato da una tanica la quale fornisce l'ingresso per la piscina, e si assume che l'acqua di ritorno sia depurata. Si intuisce che questo sistema é l'estensione di crazy river (semplificato per i calcoli) con l'introduzione di un disturbo (la pioggia).

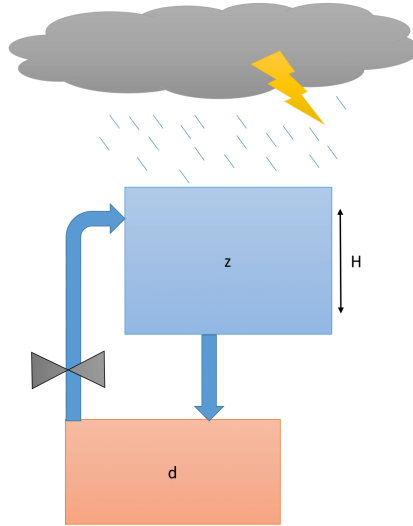


Figure 18: Sistema Scotland's Issue

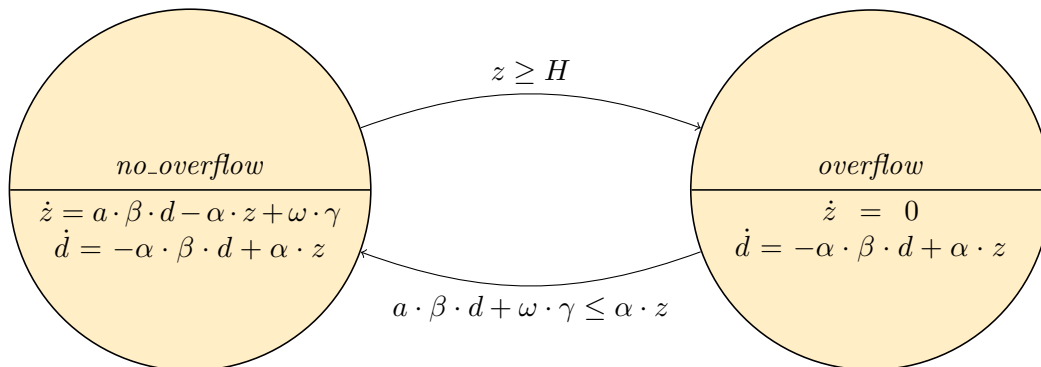
## Impianto delle taniche

Per modellare il sistema delle due taniche abbiamo bisogno di due locazioni e due variabili:

- Locazione *no\_overflow*: rappresenta la mancanza di overflow sotto una determinata soglia
- Locazione *overflow*: rappresenta il superamento della soglia e quindi lo spreco di acqua
- Variabile *z*: la tanica rappresentate la piscina
- Variabile *d*: la tanica rappresentate il depuratore (clorinatore)

Le transizioni da *no\_overflow* ed *overflow* sono dettate dalle seguenti guardie:

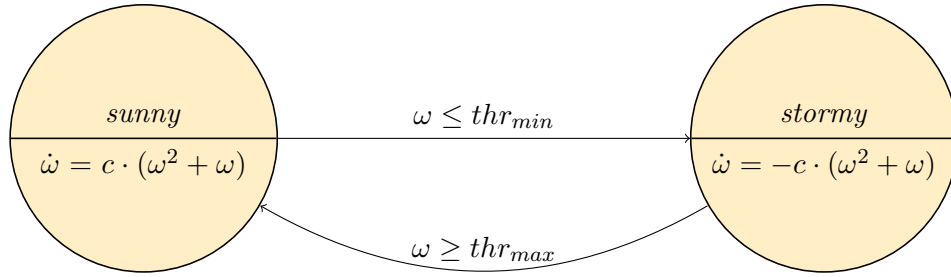
- $z \geq H$ : identifica il superamento della soglia e quindi l'inizio dell'*overflow*
- $\alpha \cdot \beta \cdot d + \omega \cdot \gamma \leq \alpha \cdot z$ : rappresenta il fatto che il flusso in ingresso é minore del flusso in uscita e quindi l'*overflow* é finito





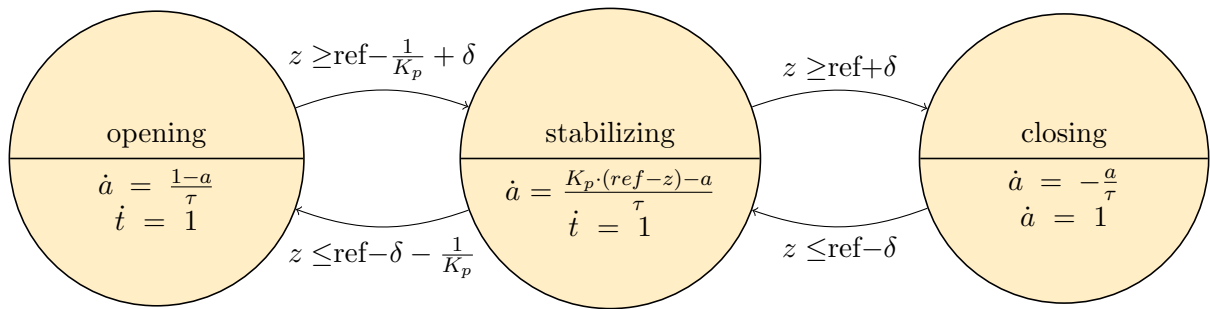
## La pioggia come disturbo

La pioggia é stata modellata usando un'unica variabile in uscita che é il valore di quanto sta piovendo. Questo sistema viene calcolato come un sistema non lineare la cui equazione é descritta da



## Controllore

É stato scelto un controllo proporzionale poiché molto più flessibile in situazioni di disturbo.



## Analisi

Il controllo proporzionale utilizzato in termini computazionali é molto pesante quando il controllore si trova nella fase di *stabilizing*. Pertanto é visibile nella figura [19a] l'espansione degli stati quando il controllore raggiunge tale fase. Questo comportamento é confermato anche nelle figure [19b] e [19c], limitando la fase di *stabilizing* (figura [19d]), imponendo un'uscita molto piccola da un certo istante in poi si nota che non vi é una espansione eccessiva degli stati raggiungibili e quindi il sistema entra in equilibrio compensando l'uscita con il disturbo della pioggia.

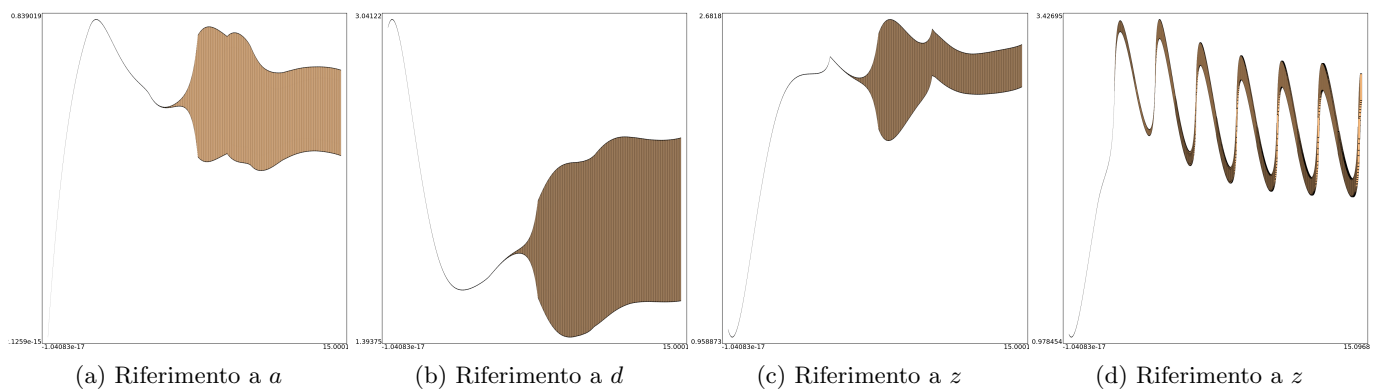


Figure 19: Evoluzione di 15 secondi del sistema

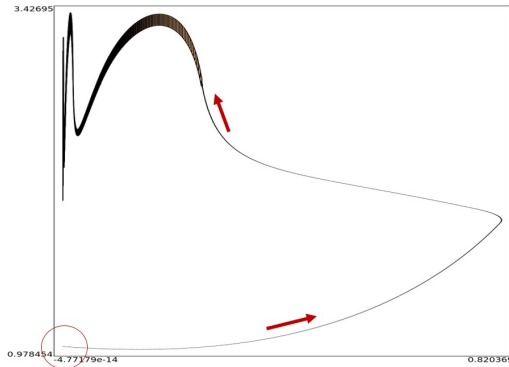
## Proporzionale vincolato

In tale settaggio come già accennato il controllo cerca di evitare la fase di stabilizzazione avendo come parametri i seguenti:

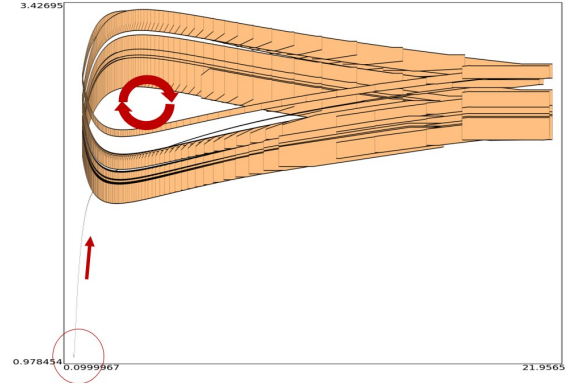
- *Initail box*:  $\{a = [0.0, 0.0], d = [3.0, 3.0], w = [0.1, 0.1], z = [1.0, 1.0]\}$

- *Parametri*:  $\{\alpha = 0.4, \beta = 0.8, \gamma = 0.4, \delta = 0.1, \tau = 1.2, H = 4.0, K_p = 10.0, thr_{max} = 20.0\}$

Infatti nella figura [20a] si nota chiaramente che l'apertura della valvola va in equilibrio con il sistema chiudendosi dopo aver portato il valore dentro il range di riferimento. Più nel dettaglio relazionando  $w$  con



(a) Relazione tra  $a$  e  $z$



(b) Relazione tra  $w$  e  $z$

Figure 20: Evoluzione di 15 secondi del sistema

$z$  (figura [20b]) si nota che il sistema é in equilibrio rispetto alla variabile  $z$  dove gli stati raggiungibili sono dentro le specifiche del problema.

### Proporzionale non vincolato

In questo caso il controllore non é limitato e quindi entra nella fase di stabilizing mantenendo l'acqua entro un *range* accettabile come certificato dalla figura [21]. É evidente che vi é una regione di convergenza nel quale il sistema computa, tuttavia i calcoli sono troppo onerosi (dovuto all'equazione che caratterizza lo stato di *stabilizing*) e quindi le traiettorie sono molto fitte a causa del passo di integrazione molto basso.

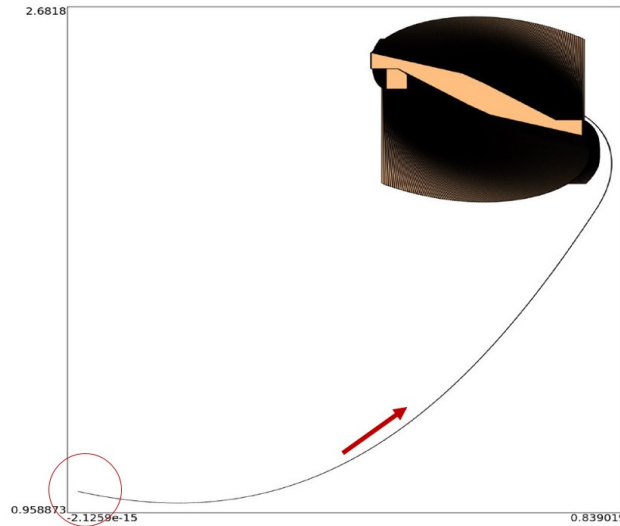


Figure 21: Relazione tra  $a$  e  $z$

## Conclusioni e considerazioni

Ariadne si é dimostrato un tool efficace ed utile per una analisi formale dei sistemi creati. Tuttavia rimane una riserva sulla *safety verification* la quale é troppo onerosa dal punto di vista computazionale per essere eseguita. Nonostante ciò ci ha permesso di analizzare i sistemi in maniera rigorosa nel rispetto delle specifiche da noi imposte, traendo delle conclusioni sulle proprietà dei sistemi come la stabilità, il rispetto dei *constraints* e i limiti. Inoltre Ariadne ci ha consentito la verifica di sistemi ibridi non lineari, come nell'esempio di Scotland's issue.