"If I cannot draw it, I do not understand it"
–Anonymous.

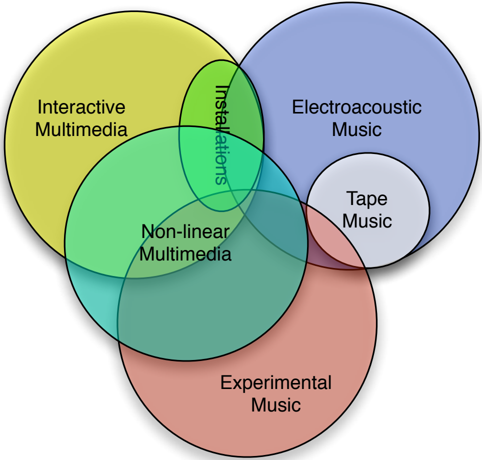
Structured interactive scores: From a structural description of a multimedia scenario to a real-time capable implementation with formal semantics

Mauricio TORO – LaBRI, Université de Bordeaux.
Ph.D defense

Supervised by Myriam Desainte-Catherine and Camilo Rueda

September 25th 2012

Some domains shaped by the arose of technology



Introduction

1. **Examples of multimedia interaction**
2. Problems with most existing tools
3. Solution: interactive scores
4. History of interactive scores
5. Tools for interactive scores
6. Related formalisms to interactive scores
7. Related tools to interactive scores

Contemporary Dance

./videos/theatre.mov

Minchoul synchronizes human gestes, video and sound in real-time in *cleaning (2009)*.

Interactive multimedia installations

./videos/installation.mov

Torres uses the participation of the audience to co-author the meaning of the installation in *pollen (2008)*.

Meta-instruments for physically disabled people

./videos/baopao.mov

The *Bao-Pao*[1] is a meta-instrument. As an example, this meta-instrument can control the start and end of notes.

[1] http://www.bao-pao.com/index.php

Introduction

1. Examples of multimedia interaction
2. **Problems with most existing tools**
3. Solution: interactive scores
4. History of interactive scores
5. Tools for interactive scores
6. Related formalisms to interactive scores
7. Related tools to interactive scores

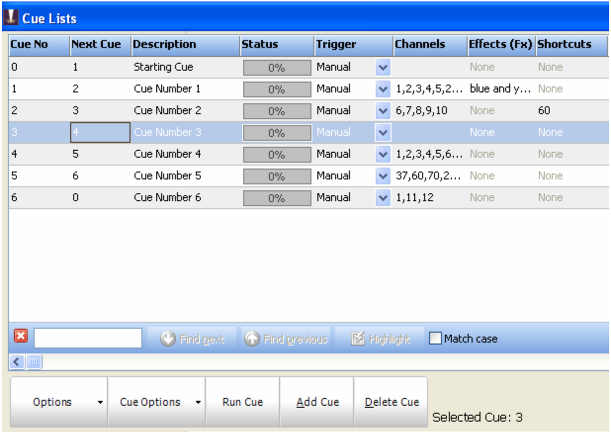
Problems with most existing tools

1. No formal semantics [TDCR12]

Problems with most existing tools

1. No formal semantics [TDCR12]
2. Time models are unrelated [DCAA12]

Cue-Lists model

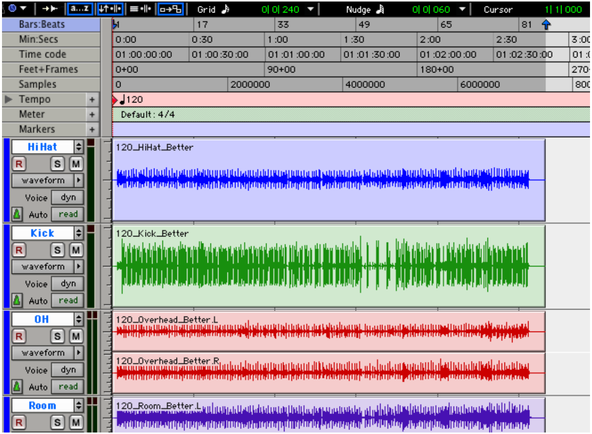


Cue No	Next Cue	Description	Status	Trigger	Channels	Effects (Fx)	Shortcuts
0	1	Starting Cue	0%	Manual		None	None
1	2	Cue Number 1	0%	Manual	1,2,3,4,5,2...	blue and y...	None
2	3	Cue Number 2	0%	Manual	6,7,8,9,10	None	60
3	4	Cue Number 3	0%	Manual		None	None
4	5	Cue Number 4	0%	Manual	1,2,3,4,5,6...	None	None
5	6	Cue Number 5	0%	Manual	37,60,70,2...	None	None
6	0	Cue Number 6	0%	Manual	1,11,12	None	None

As an example, consider theater cue-list manager Qlab [2].

[2] http://figure53.com/qlab/

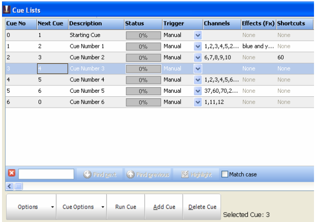
Timeline model



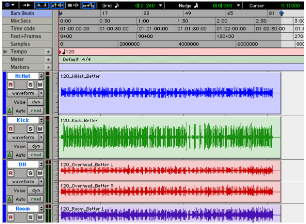
As an example, consider sequencer *Pro Tools* [3].

[3] http://www.avid.com/US/resources/digi-orientation

Time models are unrelated: Cue Lists Vs. Timeline



Vs.



It is argued in [DCAA12] that having both time models temporally related will bring new possibilities for the creation of interactive multimedia.

Problems with most existing tools

1. No formal semantics [TDCR12]
2. Time models are unrelated [DCAA12]
3. Schedulers are not appropriate for real-time [TDCC12]

Scheduler problem: aggravated under high CPU load

- Most tools do not allow the users to interact with them without letting them experience noticeable delays
- Karplus-Strong is a well-known digital signal processing algorithm to simulate the sound of metallic strings
- A Karplus-Strong arpeggio implementation in Pure Data[4].
  `./sounds/barry-loaded.mov`
- A Karplus-Strong arpeggio implementation using Faust ([OFL04]) and interactive scores [TDCC12].
  `./sounds/faust-loaded.mov`

Results from [TDCC12].

[4] Colin Barry's implementation found at www.loomer.co.uk

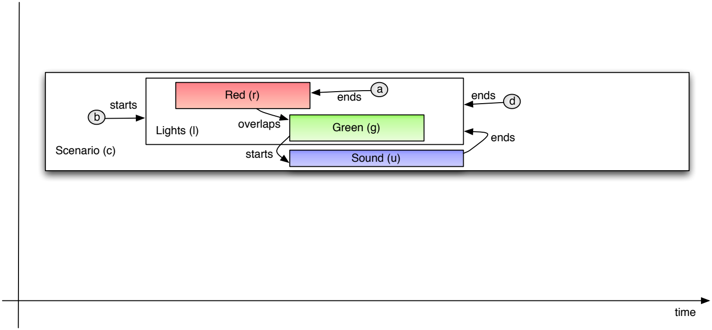
Problems with most existing tools

1. No formal semantics [TDCR12]
2. Time models are unrelated [DCAA12]
3. Schedulers are not appropriate for real-time [TDCC12]
4. No hierarchy (e.g., movements, parts, measures, motives and notes) [Vic04]
5. No unified model for conditional branching and temporal relations [Vic03, Ran09, All09, Vic11]

Introduction

1. Examples of multimedia interaction
2. Problems with most existing tools
3. **Solution: interactive scores**
4. History of interactive scores
5. Tools for interactive scores
6. Related formalisms to interactive scores
7. Related tools to interactive scores

Solution: Interactive Scores



A formalism for interactive multimedia developed at LaBRI: [BDC01, DCB03, ADC05, AADCR06, AADC07, ADCLA08, AADC08, Ran09, All09, OR09, BAM$^+$09, Sar08, ABM$^+$10], [TDCB10, TDC10, Tor10], [MADC11], [ADCT11, TDCC12, TDCR12, Tor12]

Introduction

1. Examples of multimedia interaction
2. Problems with most existing tools
3. Solution: interactive scores
4. **History of interactive scores**
5. Tools for interactive scores
6. Related formalisms to interactive scores
7. Related tools to interactive scores

History of interactive scores

- *Boxes*: A software to model a hierarchy and temporal constraints [BDC01]
- A model of interactive scores [ADC05]
- Hierarchical Time Stream Petri nets (HTS petri nets) semantics for interactive scores [AADC07]
- An extension of interactive scores with temporal reductions [ADCLA08]

History of interactive scores

- First *non-deterministic timed concurrent constraint (ntcc)* model of interactive scores [AADCR06]
- A model change the hierarchy of the temporal objects during execution [OR09]
- A model to represent conditional branching and temporal relations separately [Ran09, All09]

Introduction

1. Examples of multimedia interaction
2. Problems with most existing tools
3. Solution: interactive scores
4. History of interactive scores
5. **Tools for interactive scores**
6. Related formalisms to interactive scores
7. Related tools to interactive scores

Existing tools for interactive scores

- Boxes [BDC01]
- Iscore developed for OpenMusic [AADC08]
- Acousmouscribe
- Stand-alone i-score (currently maintained by LaBRI)
- Prototypes using Ntccrt, a real-time interpreter for ntcc [TDCB10, TDC10, TDCC12]
- Virage [BAM+09, ABM+10, MADC11]

Virage: An implementation of interactive scores

./videos/virage.mov

Software Virage developed under Virage project [BAM+09].

Introduction

1. Examples of multimedia interaction
2. Problems with most existing tools
3. Solution: interactive scores
4. History of interactive scores
5. Tools for interactive scores
6. **Related formalisms to interactive scores**
7. Related tools to interactive scores

Related formalisms to interactive scores

- Globally Asynchronous, Locally Synchronous (GALS) [TGL07]
- Hierarchical time stream Petri nets (HTS petri nets) [SSSW95]
- Temporal constraints: qualitative [All83], quantitative [Gen98] and combination of both [Mei96]
- Concurrent constraint programming (CCP) calculi [Sar92, SJG94, Sar08, OV08].
    - Non-deterministic timed concurrent constraint programming (ntcc) [NPV02]

Non-deterministic Timed Concurrent Constraint Programming (*ntcc*)

- Process calculus to model reactive systems with discrete time, non-determinism, partial information and asynchrony [NPV02, Val02]
- Includes a temporal logic to verify properties [NPV02]
- A logic formula can be translated into a process and the *strongest post-condition*[5] of a process into a *Büchi automaton*[6] [Val05]

---

[5] Sequences that the process can output under any environment.

[6] A Büchi automaton recognizes sequences of infinite length.

Non-deterministic Timed Concurrent Constraint Programming (*ntcc*)

- Has been used for the verification and simulation of other interactive multimedia systems [RV02, RV05, RV04]
- Has been used for interactive scores [AADCR06, Sar08, OR09]

Introduction

1. Examples of multimedia interaction
2. Problems with most existing tools
3. Solution: Interactive scores
4. History of interactive scores
5. Tools for interactive scores
6. Related formalism to interactive scores
7. **Related tools to interactive scores**

Related tools to interactive scores

- Score following (e.g., *Antescofo* [Con08, ECGJ11])

An example of score following

./videos/antescofo.mov

Example of the score following system *Antescofo*.

Related tools to interactive scores

- Score Following (e.g., *Antescofo* [Con08, ECGJ11])
- Asynchronous Dataflow Languages (e.g., *Max*/MSP [PAZ98] and Pure Data[Puc96])
- Synchronous Dataflow Languages (e.g., *CSound*, *Faust* [OFL04], Esterel, Signal [GLGB87] and Lustre)
- Machine Improvisation (e.g., *OMax* [ABC$^+$06] and *Continuator* [Pac02])
- Computer Assisted Composition (e.g., *OpenMusic's Maquettes* [BAA11])

Introduction

1. We saw some examples of multimedia interaction
2. There are problems with most existing tools
3. Our solution is interactive scores (IS)
4. IS is a mature formalism
5. There are several tools for IS
6. Score following and Maquettes are related to IS
7. GALS, HTSPN and ntcc are related to IS

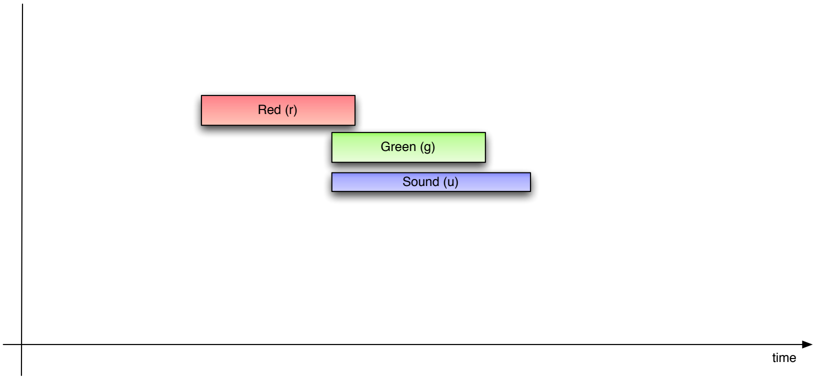
Outline of the presentation

1. **Hierarchical Model**
2. Time Conditional Model
3. Signal Processing Model
4. Simulation
5. Verification
6. Summary and Conclusions

Hierarchical model of interactive scores

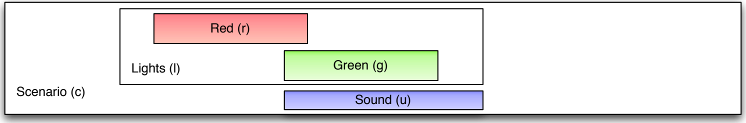
1. **Structural definition [TDCR12]**
2. Abstract semantics [TDCR12]
3. Operational semantics [TDCR12]

Structural definition: Temporal objects



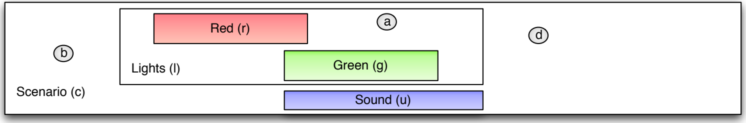
Temporal objects are aligned on a timeline, thus they have *nominal start time* and *nominal duration*.

Structural definition: Hierarchy of temporal objects



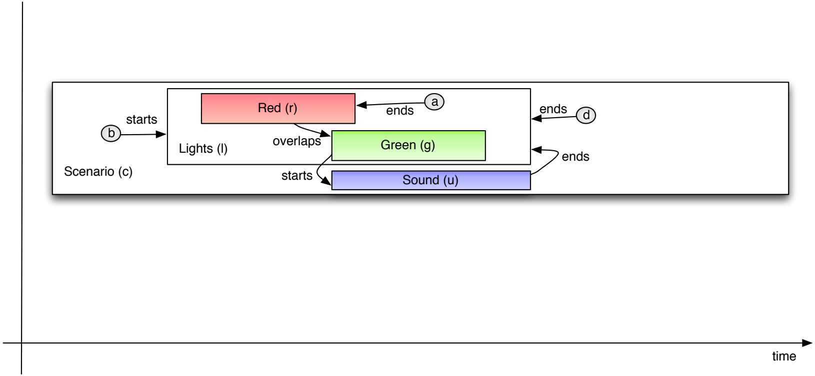
A temporal object can contain other objects to constraint the execution start and end times of its children. The hierarchical model has been previously discussed in [BDC01, All09]. We presented a new formalization in [TDCR12].

Structural definition: Interactive objects



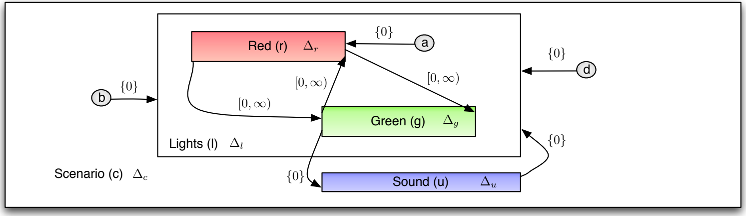
Interactive objects (gray circles) allow to control the start or end of other temporal objects.

Structural definition: Temporal relations



Temporal relations constraint the set of possible execution times and durations of the temporal objects.

Structural definition



Durations and temporal relations can be represented by point-to-point temporal relations labeled by integer intervals [Mei96].

Hierarchical model of interactive scores

1. Structural definition
2. **Abstract semantics**
3. Operational semantics

Timed Event Structures (TES)

- Langerak's *timed event structures* is a mathematical model to represent systems with non-determinism, real-time and concurrency [BKL98].

Timed Event Structures (TES)

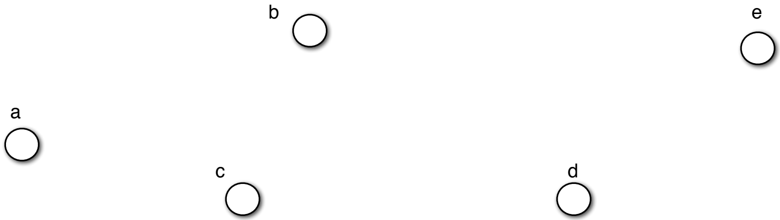


Figure : Circles are events.

Timed Event Structures (TES)

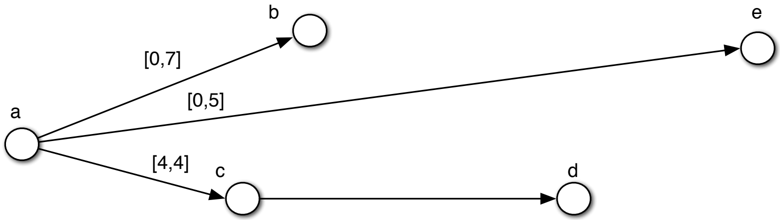


Figure : Circles are events. Full-line arrows represent event delays.

Timed Event Structures (TES)

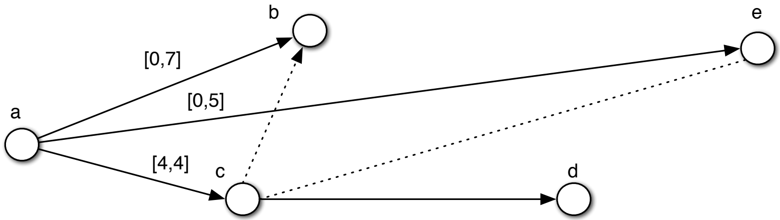
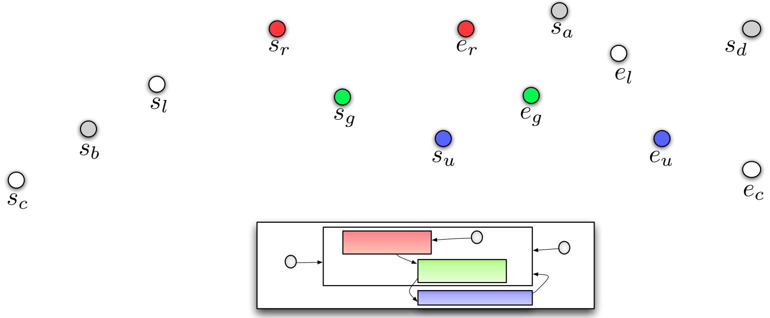


Figure : Circles are events. Full-line arrows represent event delays. Pointed-line arrows represent conflicts; when they have no direction, the conflict is symmetrical.

TES semantics: Temporal objects



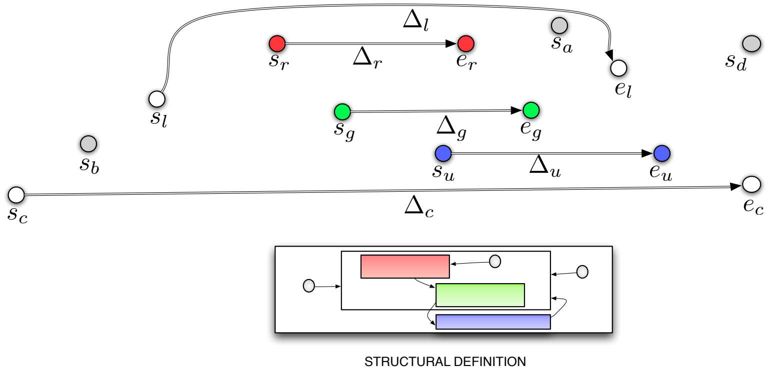
STRUCTURAL DEFINITION

TES semantics for interactive scores [TDCR12].

Colored circles represent events of static temporal objects.

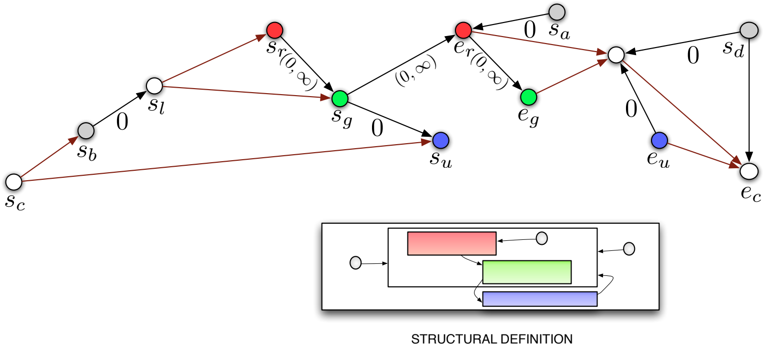
Gray circles represent events of interactive objects.

TES semantics: Duration of temporal objects



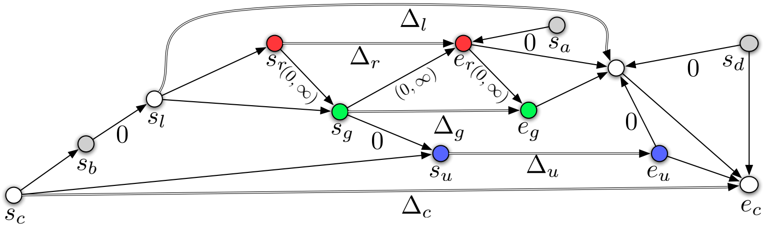
Durations of the temporal objects are represent by event delays.

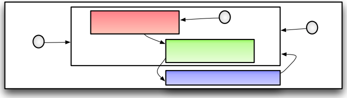
TES semantics: Temporal relations



Red arrows are event delays of the form $e_i \overset{[0,\infty)}{\mapsto} e_j$, derived from the hierarchy. Black arrows are event delays derived from the explicit temporal relations.

TES semantics: Interactive scores





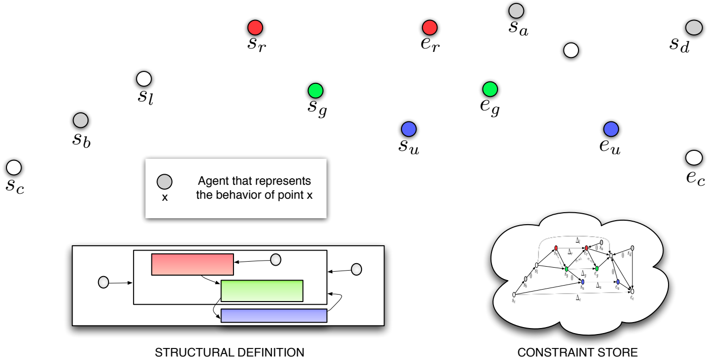
STRUCTURAL DEFINITION

TES semantics is obtained by combining the events, the event delays from the duration of temporal objects, and the event delays from the temporal relations.

Hierarchical model of interactive scores

1. Structural definition
2. Abstract semantics
3. **Operational semantics**

Operational semantics: Intuition



- Ntcc semantics for interactive scores were introduced in [AADCR06] and later formalized in [TDCR12].
- Operational semantics respects the event structures semantics

Hierarchical model of interactive scores

1. The structural definition is composed by temporal objects and point-to-point temporal relations
2. Abstract semantics are defined in timed event structures
3. Operational semantics are defined in ntcc

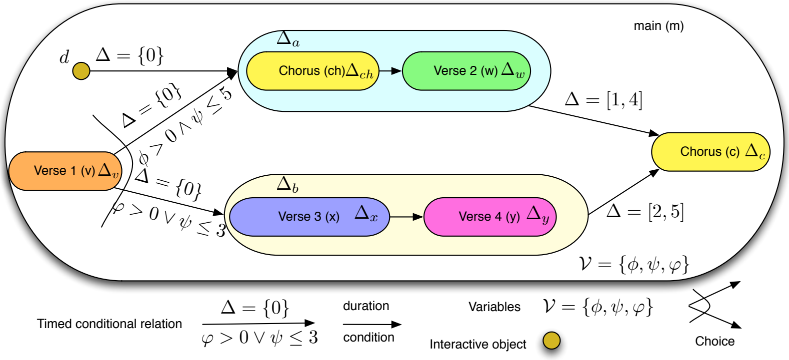
Outline of the presentation

1. Hierarchical Model
2. **Time Conditional Model**
3. Signal Processing Model
4. Simulation
5. Verification
6. Summary and Conclusions

Time conditional model of interactive scores

1. **Structural definition** [TDCB10, TDC10, ADCT11]
2. Abstract semantics
3. Operational semantics [TDCB10, TDC10, ADCT11]

Time conditional model: Structural definition

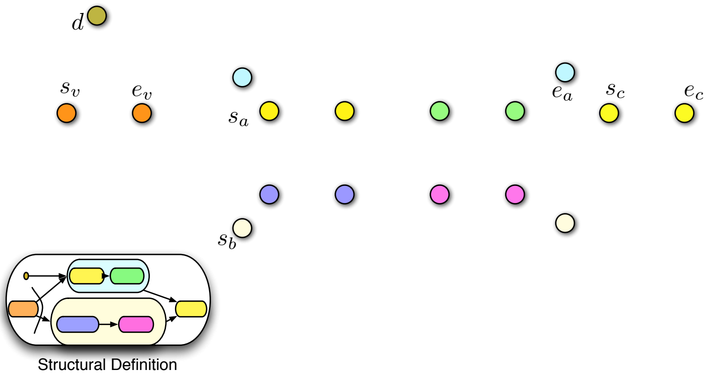


The model was introduced in [TDCB10, TDC10, ADCT11].

Time conditional model of interactive scores

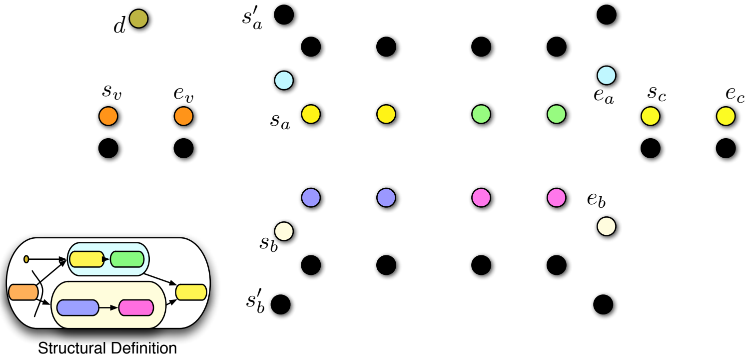
1. Structural definition
2. **Abstract semantics**
3. Operational semantics

TES semantics: "Visible"-action events



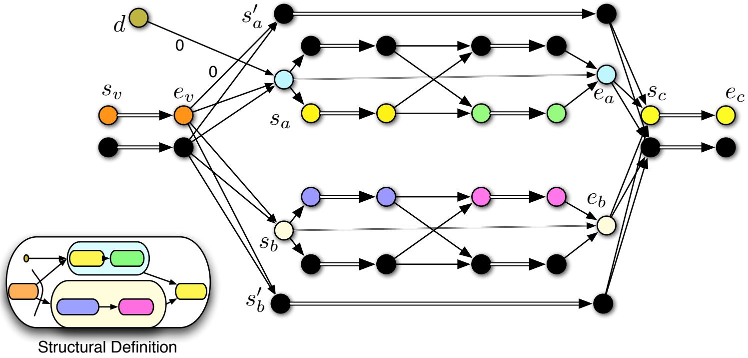
Colored circles are the "visible"-action events.

TES semantics: "Invisible"-action events



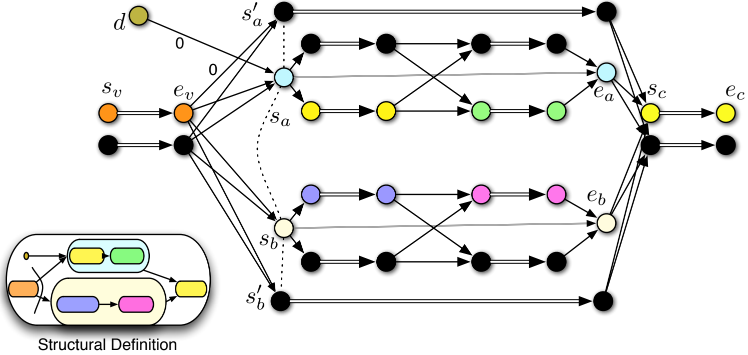
Black circles are the "invisible"-action events.

TES semantics: Temporal relations



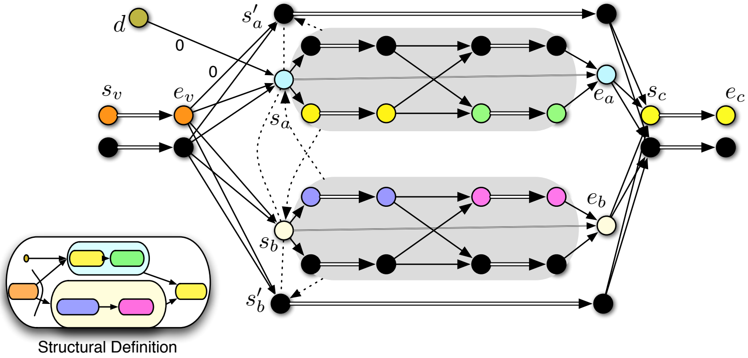
Temporal relations are translated to event delays. Event delays are represented as arrows. For simplicity, the arrows have no labels.

TES semantics: Conflicts



Dashed lines represent symmetrical conflicts: It means that two events are mutually exclusive.

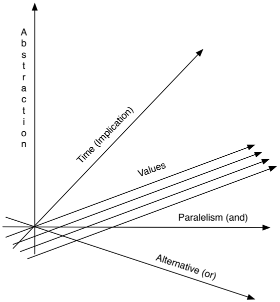
TES semantics: More conflicts



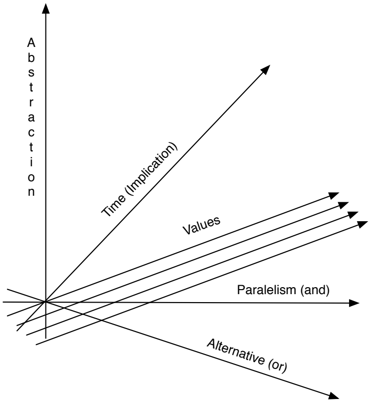
Dashed-line arrows represent asymmetrical conflicts: It means that one event must happen before another or it will not happen.

Why visible and invisible events?

- Because it is a simple way to represent conditional branching in a single bi-dimensional plane.



David Janin's dimensions



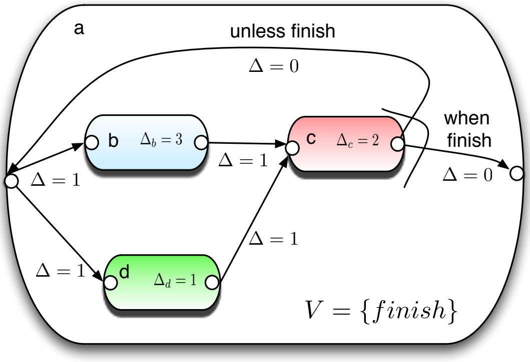
There are several dimensions in multimedia interaction[7].

[7] http://www.labri.fr/perso/janin/

Time conditional model of interactive scores

1. Structural definition
2. Abstract semantics
3. **Operational semantics**

Time conditional model: Operational semantics



Operational semantics for the conditional branching extension with loops were introduced in [TDCB10, TDC10, ADCT11].

Time conditional model of interactive scores

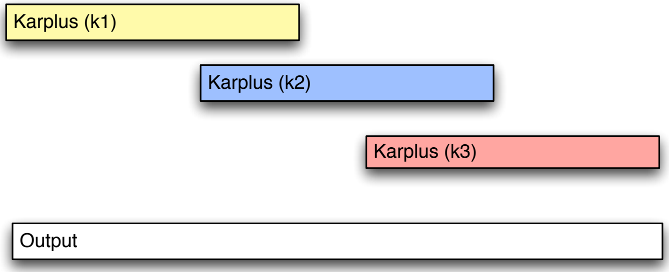
1. Structural definition is given for scores with and without loops
2. Abstract semantics are defined in timed event structures for scores without loops
3. Operational semantics are defined in ntcc, support loops, but are not related to the abstract semantics, as it was the case in the hierarchical model

Outline of the presentation

Interactive scores with signal processing

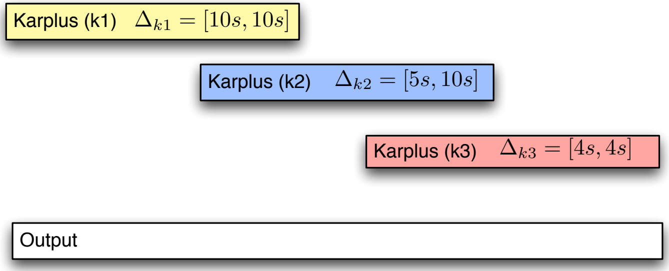
1. **Structural definition** [TDCC12]

Interactive Scores with signal processing



An extension in which ntcc and the synchronous programming language with formal semantics *Faust* [OFL04] interact. Introduced in [TDCC12] and it is inspired from [All09].

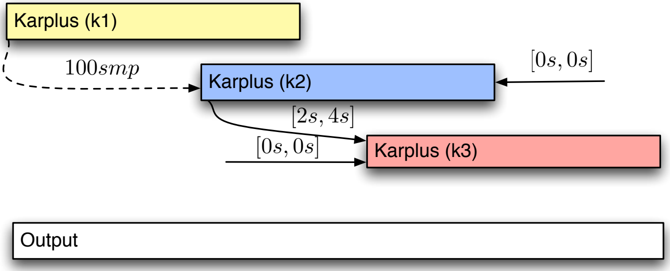
Interactive Scores: Temporal objects



Temporal objects may contain a *Faust* audio process, before they used to control external audio processes. Input[8] and Output streams are also temporal objects.

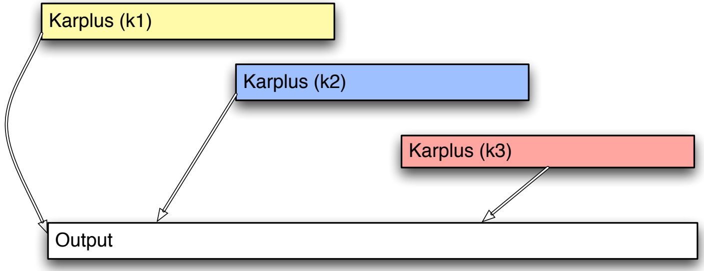
[8] In this score there is no audio input.

Interactive Scores: Temporal relations



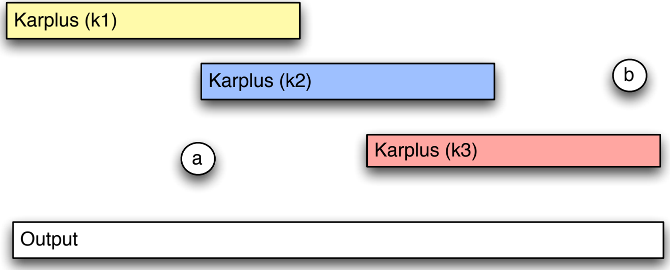
In addition to the usual temporal relations, dashed-line arrows represent *high-precision temporal relations*.

Interactive Scores: Dataflow relations



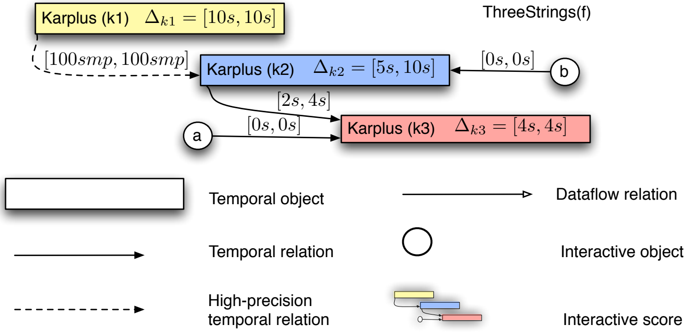
White-headed arrows represent dataflow relations: To describe how the sound is transferred from one object to another.

Interactive Scores: Interactive objects



Interactive objects are represented (and behave) as usual.

Extension of Interactive Scores



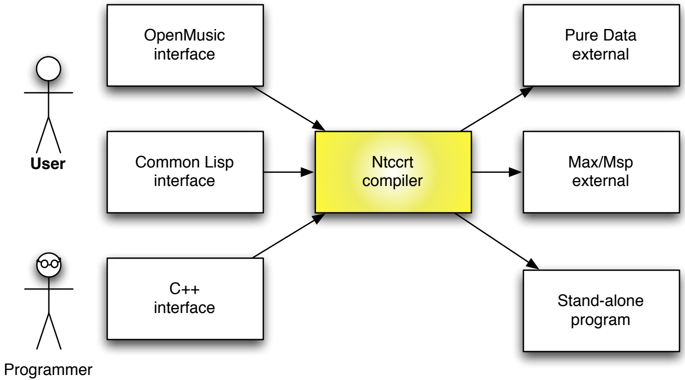
Outline of the presentation

1. Hierarchical Model
2. Time Conditional Model
3. Signal Processing Model
4. **Simulation**
5. Verification
6. Summary and Conclusions

Simulation of Interactive Scores

1. **Ntccrt** [TAAR09]
2. Results on the conditional branching model [TDC10]
3. Results on the signal processing extension [TDCC12]

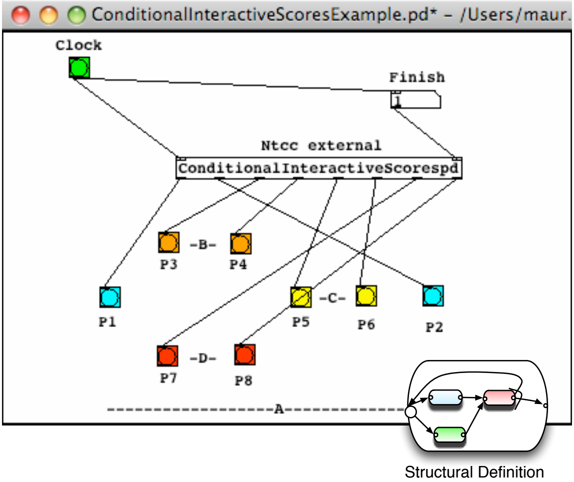
Simulation: Interpreter for ntcc



Ntccrt[9] is a simulation tool for ntcc introduced in [TAAR09].
Experience on the usage of Ntccrt is described in [ORS+11].

[9] http://sourceforge.net/projects/ntccrt/

Simulation: Example using Pure Data

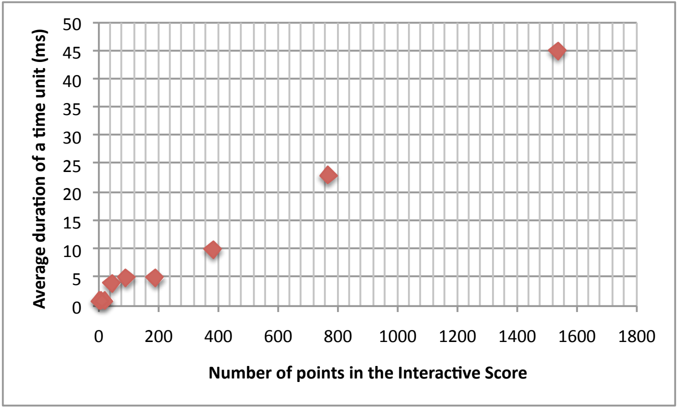


“Bang” objects are controlled by the Ntccrt plugin [TDC10].

Simulation

1. Ntccrt
2. **Results on the conditional branching model**
3. Results on the signal processing extension

Results on the conditional branching extension



Stress test to compute the average duration of a discrete time unit of the simulation of a conditional branching score using Ntccrt [TDC10].

Simulation

1. Ntccrt
2. Results on the conditional branching model
3. **Results on the signal processing extension**

Results on the signal processing extension

Average relative jitter: The average time difference between the expected starting time of each string, with respect to the first string of an arpeggio, and the time obtained during execution.

Results on the signal processing extension

Implementation	85% CPU load	3% CPU load
Using object *block*˜ 1 in Pd	7991 ms	1 ms
Using object *z*˜ 1 in Pd	9231 ms	2 ms
Using Faust object in Pd	0.5 ms	0.5 ms

Table : Average relative jitter of each string of an arpeggio with respect to the beep. All implementations use the Karplus-Strong algorithm to simulate the sound of the strings [TDCC12].

Results on the signal processing extension

In some cases –such as programs with feedback loops– our interactive scores implementation using Faust outperforms Pure Data under high-cpu load. Pure data's performance is unacceptable, in such cases, for real-time interaction.

Simulation

1. Our models are simulated using Ntccrt, a real-time capable interpreter for ntcc
2. Results on the conditional branching model show that we can execute up to 400 temporal objects
3. Results on the signal processing extension show that interactive scores outperforms Pure Data under high-cpu load

Outline of the presentation

1. Hierarchical Model
2. Time Conditional Model
3. Signal Processing Model
4. Simulation
5. **Verification**
6. Summary and Conclusions

Verification

1. **Motivation**
2. Constraint linear-time temporal logic (CLTL) [NPV02]
3. Model checking algorithm

Verification of interactive scores

- Existing semantics for interactive scores were given in HTS petri nets.
- Existing verification tools for time Petri nets, such as Roméo [GLMR05] and Tapaal [BJS09], cannot be used to verify HTS petri nets [SSSW95] because HTS petri nets cannot be translated to time Petri nets [BD99].
- Existing verification tools for linear-time logic, such as : Uppaal [BDL+01], Kronos [BDM+98] and Spin [Hol97], cannot be used directly for ntcc.

Verification of interactive scores

- Results on bisimilarity for ccp [Ari10, ABPV12] are quite promising, but not yet applicable to ntcc.
- Strongest post-condition of a process can be translated into a Büchi automaton [Val05], but using classic Schimpf's algorithm [SMS09] is intractable because the encoding heavily relies on the complement of such automata [Saf88].
- We proposed a bounded-time finite-state-automata-based model checker for ntcc, *ntcc*MC[10].

[10] http://sourceforge.net/projects/ntccmc/

Verification

1. Motivation
2. **Constraint linear-time temporal logic** (CLTL)
3. Model checking algorithm

CLTL properties

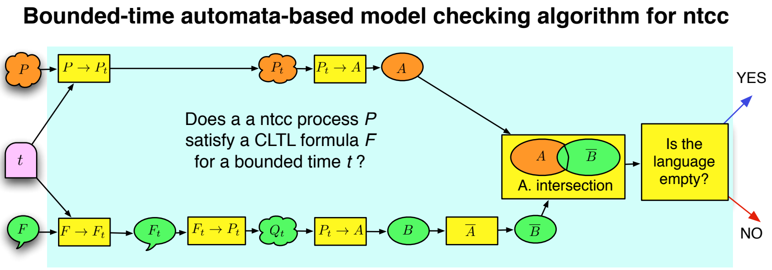
Examples of *constraint linear-time logic* (CLTL) [NPV02] formulae are

- "always the constraint $pitch = 60$ can be deduced from the output store", namely $\square\ (pitch = 60)$; and
- "eventually, both object a and object b are launched at the same time", namely $\diamond(launch_a \dot{\wedge} launch_b)$.

Verification

1. Motivation
2. Constraint linear-time temporal logic (CLTL)
3. **Model checking algorithm**

Automatic verification of ntcc



Adaptation of classic model checking algorithm for linear-time logic [SMS09] to ntcc, based on the encoding of ntcc to Büchi automata given by Valencia *et al.* [Val05].

Automatic verification of ntcc

Bounded-time automata-based model checking algorithm for ntcc

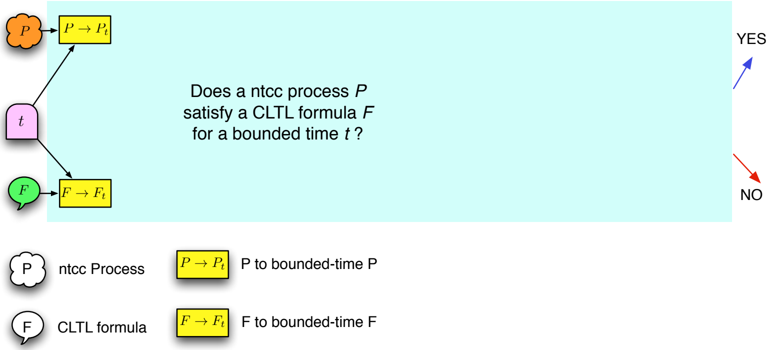
P ntcc Process

t Number of discrete time units

F CLTL formula

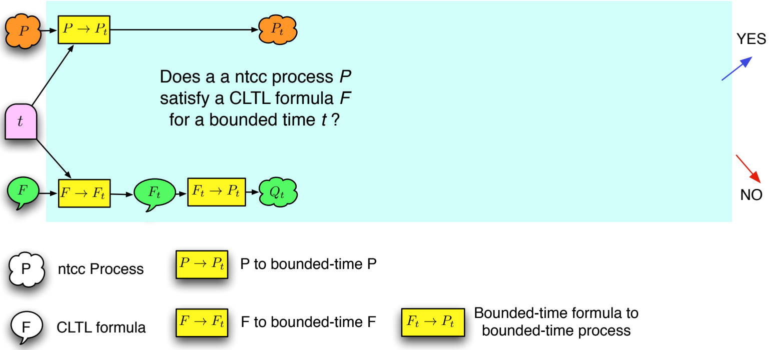
Automatic verification of ntcc

Bounded-time automata-based model checking algorithm for ntcc



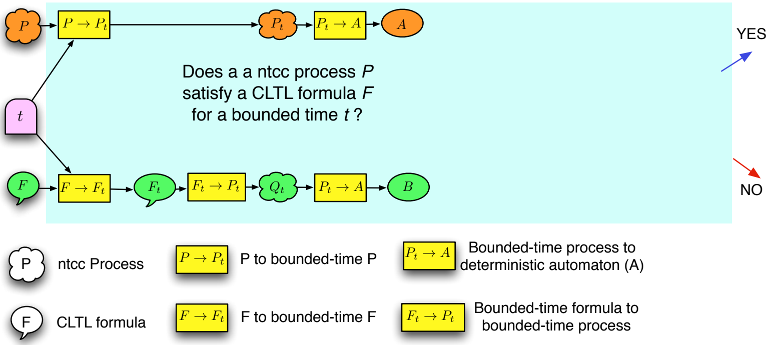
Automatic verification of ntcc

Bounded-time automata-based model checking algorithm for ntcc



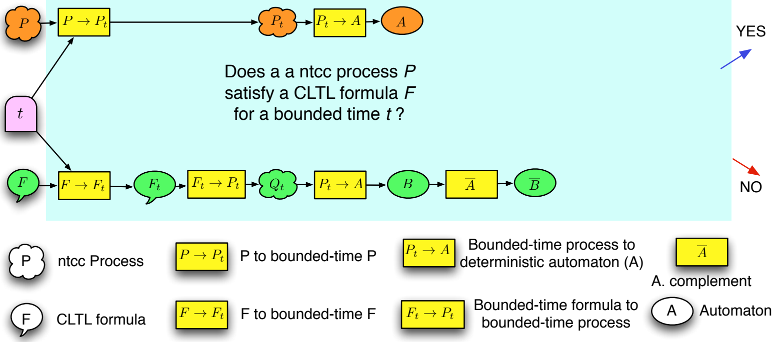
Automatic verification of ntcc

Bounded-time automata-based model checking algorithm for ntcc



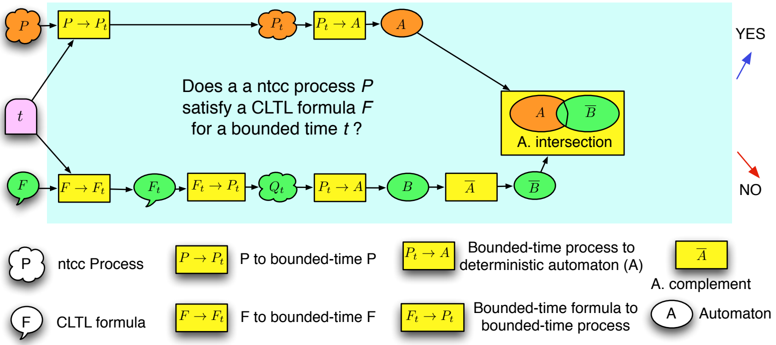
Automatic verification of ntcc

Bounded-time automata-based model checking algorithm for ntcc

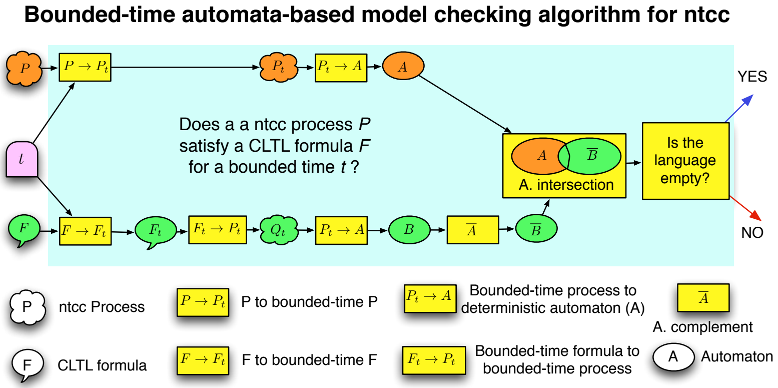


Automatic verification of ntcc

Bounded-time automata-based model checking algorithm for ntcc



Automatic verification of ntcc



Verification

1. Existing tools cannot be used directly to verify interactive scores and we want to prove properties of interactive scores automatically
2. In particular, constraint linear-time temporal logic (CLTL) properties
3. We presented our bounded-time automata-based model checking algorithm for ntcc, ntccMC, for CLTL properties

Outline of the presentation

1. Hierarchical Model
2. Time Conditional Model
3. Signal Processing Model
4. Simulation
5. Verification
6. **Conclusions**

Conclusions

1. **Summary**
2. Solutions to the problem statements [TDCC12, TDCR12]
3. Comparison w.r.t previous models of interactive scores [TDCR12]
4. Comparison w.r.t *Csound, Pure Data, Max, Live, QLab and Protools* [TDCC12]
5. Future work directions
6. Ongoing projects

Summary

- A Formal model for multimedia interaction that combines
  - timeline model
  - cue-list model [TDCR12]
- An extension for conditional branching [TDCB10, TDC10, ADCT11]
- An extension for audio processing [Tor10, TDCC12]
- A simulation tool [TAAR09, ORS^{+}11]
- A verification tool [Tor10]

Contributions

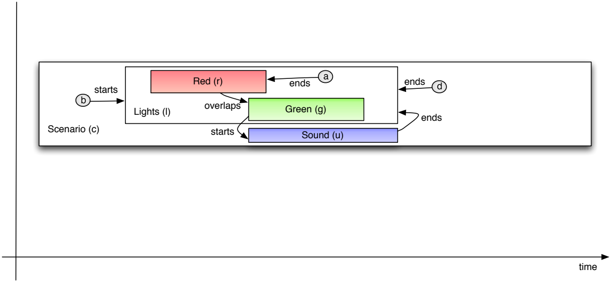
- 2 book chapters in "Constraint Programming in Music" [ADCT11, ORS+11]
- 1 journal article in "Mathematics and Music" [TDCR12]
- 2 papers in "Sound and Music Computing" conference [TDC10, TDCC12]
- 1 extended abstract in "International Conference on Logic Programming" [Tor10]
- 1 paper in "Journées d'Informatique Musicale" [TDCB10]

Conclusions

1. Summary
2. **Solutions to the problem statements [TDCC12, TDCR12]**
3. Comparison w.r.t previous models of interactive scores [TDCR12]
4. Comparison w.r.t *Csound, Pure Data, Max, Live, QLab and Protools* [TDCC12]
5. Future work directions
6. Ongoing projects

Solutions to the problem statements

- Time models are unrelated
- In interactive scores, time models are temporally related



Solutions to the problem statements

- Schedulers are not appropriate for real-time
- Using interactive scores and Faust, it is possible to have real-time interaction even under high cpu load

Solutions to the problem statements

- No hierarchy
- Interactive scores provide a hierarchy of temporal objects

Solutions to the problem statements

- No formal semantics
- We defined abstract and operational semantics for interactive scores

Solutions to the problem statements

- No unified model for conditional branching and temporal relations
- We provide a unified model for conditional branching and temporal relations in which all branches have the same set of possible durations

Conclusions

1. Summary
2. Solutions to the problem statements [TDCC12, TDCR12]
3. **Comparison w.r.t previous models of interactive scores [TDCR12]**
4. Comparison w.r.t *Csound, Pure Data, Max, Live, QLab and Protools* [TDCC12]
5. Future work directions
6. Ongoing projects

Comparison w.r.t previous models of interactive scores

- Model to change the hierarchy during performance [OR09]
- Previous model developed at LaBRI [All09]
- Our models [TDC10, TDCB10, TDCR12, TDCC12]

Model to change the hierarchy during performance

- Provides an elegant solution to change the structure of the hierarchy during performance
- Has not been implemented

Previous model developed at LaBRI

- Supports different mechanisms of temporal reductions
- Average duration of a discrete time unit outperforms our simulations
- Durations are intervals are limited to $\{0\}$, $(0, \infty]$ and $[0, \infty]$
- Conditional branching and temporal relations are presented separately
- Processes associated are meant to be handled outside the formalism; in the implementation as external programs

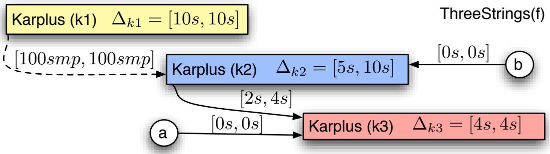
Our models

- Abstract semantics to describe the execution traces
- Arbitrary integer intervals in the operational semantics
- Unified model for conditional branching and temporal relations
- Automatic verification
- Does not allow temporal reductions
- Simulation tool is a prototype; it is not user friendly
- Average duration of time units is slower than in the implementation of previous model

Conclusions

1. Summary
2. Solutions to the problem statements [TDCC12, TDCR12]
3. Comparison w.r.t previous models of interactive scores [TDCR12]
4. **Comparison w.r.t Csound, Pure Data, Max, Live, QLab and Protools [TDCC12]**
5. Future work directions
6. Ongoing projects

Comparison w.r.t *Csound, Pure Data, Max, Live, QLab and Protools*



- Qlab and Ableton Live do not allow to model delays of 100 samples
- Protools does not allow interaction

In all the programs mentioned above, it is very hard to synchronize processes whose durations are integer intervals such as *duration* $\in [2, 4]$.

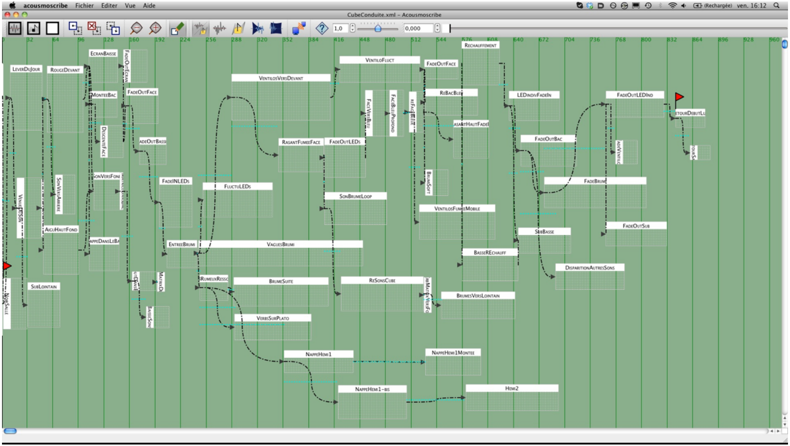
Conclusions

1. Summary
2. Solutions to the problem statements [TDCC12, TDCR12]
3. Comparison w.r.t previous models of interactive scores [TDCR12]
4. Comparison w.r.t *Csound, Pure Data Max, Live, QLab and Protools* [TDCC12]
5. **Future work directions**
6. Ongoing projects

Future work directions

- Event structures semantics for scores with loops
- Model interactive theater performances
- Model interactive museum exhibitions
- Verify scores in which ntcc and Faust interact
- Model music pieces combining the timeline and cue-list model

Music pieces combining both time models



Baltazar *et al.* used *i-score* to compose *miniature # 1 (2012)*.
I-score controls the smoke, fan, sound and lights.

Music pieces combining both time models

./videos/pascal.mp4

Baltazar *et al.* needed to have objects with flexible duration because the behavior of the smoke is unpredictable.

Conclusions

1. Summary
2. Solutions to the problem statements [TDCC12, TDCR12]
3. Comparison w.r.t previous models of interactive scores [TDCR12]
4. Comparison w.r.t *Csound, Pure Data, Max, Live, QLab and Protools* [TDCC12]
5. Future work directions
6. **Ongoing projects**

Ongoing projects

- Project *Interactivity in the Writing of Time and Interactions* (INEDIT) (2012-2015) founded by the french *national research agency* (ANR).
- Project *Robust theories for Emerging Applications in Concurrency Theory: Processes and Logic Used in emergent Systems*(REACT+) (2011-2013) founded by the colombian national research agency (*Colciencias*).
- Project OSSIA (2012-2015) founded by the ANR.

Ongoing projects

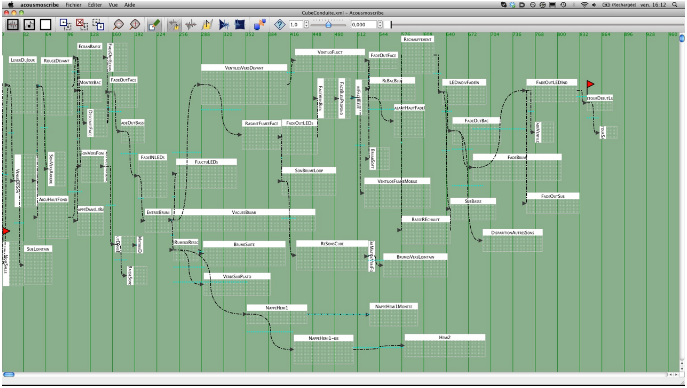
- Project *Interactivity in the Writing of Time and Interactions* (INEDIT) (2012-2015) founded by the french *national research agency* (ANR).
- Project *Robust theories for Emerging Applications in Concurrency Theory: Processes and Logic Used in emergent Systems*(REACT+) (2011-2013) founded by the colombian national research agency (*Colciencias*).
- Project OSSIA (2012-2015) founded by the ANR.

Ongoing projects

- Project *Interactivity in the Writing of Time and Interactions* (INEDIT) (2012-2015) founded by the french *national research agency* (ANR).
- Project *Robust theories for Emerging Applications in Concurrency Theory: Processes and Logic Used in emergent Systems*(REACT+) (2011-2013) founded by the colombian national research agency (*Colciencias*).
- Project OSSIA (2012-2015) founded by the ANR.

M E R C I.

Avez-vous des questions?



Outline of the presentation

1. Hierarchical Model
2. Time Conditional Model
3. Signal Processing Model
4. Simulation
5. Verification
6. Summary and Conclusions
7. **Extra**

Antoine Allombert, Gérard Assayag, and Myriam Desainte-Catherine.
A system of interactive scores based on petri nets.
In *Proc. of SMC '07*, Athens, Greece, 2007.

Antoine Allombert, Gérard Assayag, and Myriam Desainte-Catherine.
Iscore: a system for writing interaction.
In *Proc. of 3rd International Conference on Digital Interactive Media in Entertainment and Arts (DIMEA) '08*, pages 360–367, New York, NY, USA, 2008. ACM.

Antoine Allombert, Gérard Assayag, M. Desainte-Catherine, and Camilo Rueda.
Concurrent constraint models for interactive scores.
In *Proc. of Sound and Music Computing (SMC) '06*, Marseille, France, 2006.

Gérard Assayag, Georges Bloch, Marc Chemillier, Arshia Cont, and Shlomo Dubnov.
Omax brothers: a dynamic topology of agents for improvization learning.
In *Proc. of the 1st ACM workshop on Audio and music computing multimedia (AMCMM) '06*, pages 125–132, New York, NY, USA, 2006. ACM.

Antoine Allombert, Pascal Baltazar, Raphaël Marczak, Myriam Desainte-Catherine, and Laurent Garnier.
Designing an interactive intermedia sequencer from users requirements and theoretical background.
In *Proc. of International Computer Music Conference (ICMC) '10*, 2010.

A. Aristizabal, F. Bonchi, L. Pino, and F. Valencia.
Partition refinement for bisimilarity in ccp.

In *Proc. of the 27th ACM Symposium On Applied Computing (SAC 2012)*, 2012.

Antoine Allombert and Myriam Desainte-Catherine.
Interactive scores: A model for specifying temporal relations between interactive and static events.
In *Journal of New Music Research*, 2005.

Antoine Allombert, Myriam Desainte-Catherine, J. Larralde, and Gérard Assayag.
A system of interactive scores based on qualitative and quantitative temporal constraints.
In *Proc. of 4th International Conference on Digital Arts (Artech) '08*, Porto, Portugal, 2008. The Artech International Association.

Antoine Allombert, Myriam Desainte-Catherine, and Mauricio Toro.

Modeling temporal constrains for a system of interactive score.
In Gérard Assayag and Charlotte Truchet, editors, *Constraint Programming in Music*, chapter 1, pages 1–23. Wiley, 2011.

James F. Allen.
Maintaining knowledge about temporal intervals.
Communication of ACM, 26, 1983.

Antoine Allombert.
Aspects temporels d'un système de partitions numèriques interactives pour la composition et l'interprétation.
PhD thesis, Université de Bordeaux, November 2009.

Andrés Aristizábal.
Bisimilarity in concurrent constraint programming.
In *26th International Conference on Logic Programming (ICLP) 2010*, 2010.

Jean Bresson, Carlos Agon, and Gérard Assayag.
Openmusic: visual programming environment for music composition, analysis and research.
In *Proceedings of the 19th ACM international conference on Multimedia*, MM '11, pages 743–746, New York, NY, USA, 2011. ACM.

P. Baltazar, A. Allombert, R. Marczak, J.M. Couturier, M. Roy, A. Sèdes, and M. Desainte-Catherine.
Virage : Une reflexion pluridisciplinaire autour du temps dans la creation numerique.
In *in Proc. of JIM*, 2009.

Marc Boyer and Michel Diaz.
Non equivalence between time petri nets and time stream petri nets.

In *Proc. of the The 8th International Workshop on Petri Nets and Performance Models*, number 98 in 1, Washington, DC, USA, 1999. IEEE Computer Society.

Anthony Beurivé and Myriam Desainte-Catherine.
Representing musical hierarchies with constraints.
In *7th International Conference on Principles and Practice of Constraint Programming, Musical Constraints Workshop, Paphos*, 2001.

Gerd Behrmann, Alexandre David, Kim G. Larsen, Oliver Möller, Paul Pettersson, and Wang Yi.
UPPAAL - present and future.
In *Proc. of 40*th *IEEE Conference on Decision and Control*. IEEE Computer Society Press, 2001.

Marius Bozga, Conrado Daws, Oded Maler, Alfredo Olivero, Stavros Tripakis, and Sergio Yovine.
Kronos: A model-checking tool for real-time systems.

In Anders Ravn and Hans Rischel, editors, *Formal Techniques in Real-Time and Fault-Tolerant Systems*, volume 1486 of *Lecture Notes in Computer Science*, pages 298–302. Springer Berlin / Heidelberg, 1998.

Joakim Byg, Kenneth Yrke Jørgensen, and Jiří Srba.
Tapaal: Editor, simulator and verifier of timed-arc petri nets.
In *Proceedings of the 7th International Symposium on Automated Technology for Verification and Analysis (ATVA) '09*, pages 84–89, Berlin, Heidelberg, 2009. Springer-Verlag.

Christel Baier, Joost-Pieter Katoen, and Diego Latella.
Metric semantics for true concurrent real time.
In *Proc. of International Conference on Automata, Languages and Programming (ICALP) '98*, Berlin, Germany, 1998. Springer.

Arshia Cont.
Antescofo: Anticipatory synchronization and control of interactive parameters in computer music.
In *Proc. of ICMC '08*, 2008.

Myriam Desainte-Catherine, Antoine Allombert, and Gérard Assayag.
Towards a hybrid temporal paradigm for musical composition and performance: The case of musical interpretation.
Computer Music Journal, To appear in fall, 2012.

M. Desainte-Catherine and N. Brousse.
Towards a specification of musical interactive pieces.
In *Proc. of the XIX Colloquium on Musical Informatics (CIM)*, Firenze, Italy, 2003.

José Echeveste, Arshia Cont, Jean-Louis Giavitto, and Florent Jacquemard.
Formalisation des relations temporelles entre une partition et une performance musicale dans un contexte d'accompagnement automatique : Accompagnement musical automatique.
In *Journal Européen des Systèmes Automatisés*, 2011.

Rosella Gennari.
Temporal resoning and constraint programming - a survey.
CWI Quaterly, 11:3–163, 1998.

Thierry Gautier, Paul Le Guernic, and Löic Besnard.
Signal: A declarative language for synchronous programming of real-time systems.
In *Proc. of FPCA '87*, 1987.

Guillaume Gardey, Didier Lime, Morgan Magnin, and Olivier (H.) Roux.
Roméo: A tool for analyzing time Petri nets.
In *17th International Conference on Computer Aided Verification (CAV'05)*, volume 3576 of *Lecture Notes in Computer Science*, pages 418–423, Edinburgh, Scotland, UK, 2005. Springer.

Gerard J. Holzmann.
The model checker spin.
Software Engineering, 23(5):279–295, 1997.

Raphaël Marczak, Antoine Allombert, and Myriam Desainte-Catherine.
Real-time temporal control of musical processes.
In *Proc. of the International Conferences on Advances in Multimedia (MMEDIA) '11*, 2011.

Itay Meiri.
Combining qualitative and quantitative constraints in temporal reasoning.
Artificial Inteligence, 87(1-2):343–385, 1996.

M. Nielsen, C. Palamidessi, and F. Valencia.
Temporal concurrent constraint programming: Denotation, logic and applications.
Nordic Journal of Computing, 1(9):145–188, 2002.

Yann Orlarey, Dominique Fober, and Stephane Letz.
Syntactical and semantical aspects of faust.
Soft Comput., 8(9):623–632, 2004.

Carlos Olarte and Camilo Rueda.
A Declarative Language for Dynamic Multimedia Interaction Systems.

In *Proc. of Mathematics and Computation in Music*, volume 38, Berlin, Germany, july 2009. Springer.

Carlos Olarte, Camilo Rueda, Gerardo Sarria, Mauricio Toro, and Frank Valencia.
Concurrent constraints models of music interaction.
In Gérard Assayag and Charlotte Truchet, editors, *Constraint Programming in Music*, chapter 6, pages 133–153. Wiley, Hoboken, NJ, USA., 2011.

Carlos Olarte and Frank D. Valencia.
The expressivity of universal timed ccp: undecidability of monoadic fltl and closure operators for security.
In *PPDP '08: Proceedings of the 10th international ACM SIGPLAN conference on Principles and practice of declarative programming*, pages 8–19. ACM, 2008.

François Pachet.

Playing with virtual musicians: the continuator in practice.

IEEE Multimedia, 9:77–82, 2002.

Miller Puckette, Theodore Apel, and David Zicarelli.
Real-time audio analysis tools for pd and max/msp.
In *Proc. of ICMC '98*, Ann Arbor, USA, 1998.

M. Puckette.
Pure data.
In *Proceedings of the International Computer Music Conference. San Francisco 1996*, 1996.

Nirina Ranaivoson.
Réflexion sur la mise en place des structures logiques dans un logiciel de partitions musicales interactives.
Master's thesis, Université de Bordeaux, 2009.

Camilo Rueda and Frank Valencia.

Proving musical properties using a temporal concurrent constraint calculus.
In *Proc. of the 28th International Computer Music Conference (ICMC) '02*, 2002.

C. Rueda and F. Valencia.
On validity in modelization of musical problems by ccp.
Soft Computing, 8(9):641–648, 2004.

Camilo Rueda and Frank Valencia.
A temporal concurrent constraint calculus as an audio processing framework.
In *SMC '05*, 2005.

S. Safra.
On the complexity of omega-automata.
In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science (SFCS) '88*, pages

319–327, Washington, DC, USA, 1988. IEEE Computer Society.

Vijay A. Saraswat.
Concurrent Constraint Programming.
MIT Press, Cambridge, MA, 1992.

Gerardo Sarria.
Formal Models of Timed Musical Processes.
PhD thesis, Universidad del Valle, Colombia, 2008.

Vijay A. Saraswat, Radha Jagadeesan, and Vineet Gupta.
Foundations of timed concurrent constraint programming.
In *Proceedings of the Ninth Annual IEEE Symposium on Logic in Computer Science*, pages 71–80. IEEE Computer Press, 1994.

Alexander Schimpf, Stephan Merz, and Jan-Georg Smaus.

Construction of büchi automata for ltl model checking verified in isabelle/hol.
In *Proceedings of the 22nd International Conference on Theorem Proving in Higher Order Logics*, TPHOLs '09, pages 424–439, Berlin, Heidelberg, 2009. Springer-Verlag.

Patrick Sénac, Pierre de Saqui-Sannes, and Roberto Willrich.
Hierarchical time stream petri net: A model for hypermedia systems.
In *Proc. of the 16th International Conference on Application and Theory of Petri Nets*, pages 451–470, London, UK, 1995. Springer-Verlag.

Mauricio Toro, Carlos Agón, Gérard Assayag, and Camilo Rueda.
Ntccrt: A concurrent constraint framework for real-time interaction.

In *Proc. of ICMC '09*, Montreal, Canada, 2009.

Mauricio Toro and Myriam Desainte-Catherine.
Concurrent constraint conditional branching interactive scores.
In *Proc. of SMC '10*, Barcelona, Spain, 2010.

Mauricio Toro, Myriam Desainte-Catherine, and P. Baltazar.
A model for interactive scores with temporal constraints and conditional branching.
In *Proc. of Journées d'Informatique Musical (JIM) '10*, May 2010.

Mauricio Toro, Myriam Desainte-Catherine, and Julien Castet.
An extension of interactive scores for multimedia scenarios with temporal relations for micro and macro controls.

In *Proc. of Sound and Music Computing (SMC) '12*, Copenhagen, Denmark, July 2012.

Mauricio Toro, Myriam Desainte-Catherine, and Camilo Rueda.
Formal semantics for interactive music scores: A framework to design, specify properties and execute interactive scenarios.
Journal of Mathematics and Music, To be published., 2012.

Paul Teehan, Mark R. Greenstreet, and Guy G. Lemieux.
A survey and taxonomy of gals design styles.
IEEE Design & Test of Computers, 24(5):418–428, 2007.

Mauricio Toro.
Structured musical interactive scores (short).
In *Proc. of the doctoral consortium in ICLP '10*, Edinburgh, Scotland, UK, 2010.

Mauricio Toro.
Structured Interactive Scores: From a simple structural definition to a real-time capable implementation.
PhD thesis, Université de Bordeaux, Bordeaux, September 2012.

Frank D. Valencia.
Temporal Concurrent Constraint Programming.
PhD thesis, University of Aarhus, 2002.

Frank D. Valencia.
Decidability of infinite-state timed ccp processes and first-order ltl.
Journal of Theoretical Computer Science - Expressiveness in concurrency, 330(3):557–607, 2005.

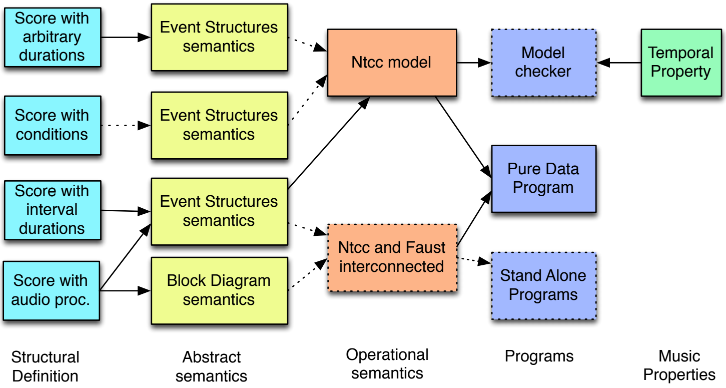
Lindsay Vickery.

Non-linear structures for real-time interactive musical works.
In *Proceedings of the Australasian Computer Music Conference (ACMC) '03*, 2003.

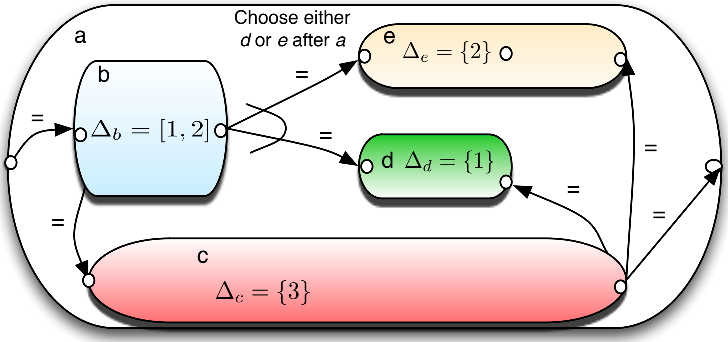
Lindsay Vickery.
Interactive control of higher order musical structures.
In *Proc. of ACMC '04*, Victoria University, New Zealand, July 2004. ACMA.

Lindsay Vickery.
The possibilities of novel format structures through computer controlled live performance.
In *Proc. of ACMC '11*, 2011.

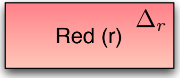
More Future Work

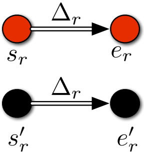


Limitation of time conditional branching

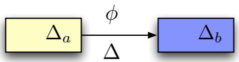


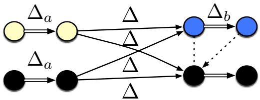
Conditional branching semantics in detail



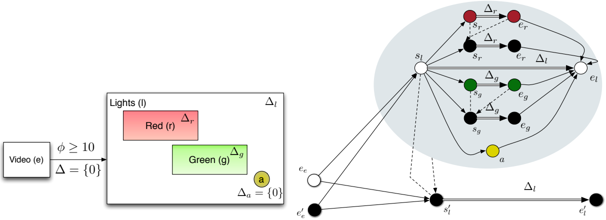


Conditional branching semantics in detail





Conditional branching semantics in detail



Equivalence between interactive score constraints and its event traces

Theorem

The temporal constraints of a score are equivalent to the temporal constraints of the event structures semantics.

Proof.

This holds by replacing points in the temporal constraints of the score by its corresponding events. □

Correctness of the operational semantics with respect to the event structures semantics

Theorem

The possible time units in which a point process can be launched in the operational semantics are included in the possible time units in which an event can appear in an event trace in the TES *semantics [TDCR12].*

Proof.

By induction on the structure of the timed event structures (TES) semantics of a score. □

Correctness of the operational semantics with respect to the event structures semantics

Theorem

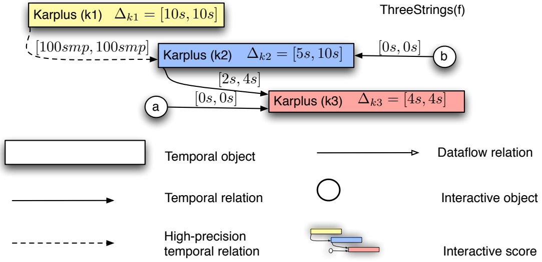
Let s be a score, $\varepsilon^* = \mathrm{des}(\mathrm{normal}(\mathrm{es}(s))) = \langle E, I^*, R \rangle$ *its event structures semantics,* $\mathrm{tc}(\varepsilon^*)$ *its temporal constraints,* $P = Score_{S,I,\mathrm{Pr},R}$ *the ntcc process that represents the score,* $[\![P]\!]$ *the denotation of the ntcc process, and* T_i *is the set of indexes* j *such that* $[\![P]\!]_{j+1}$ *entails* $launch_i$*. It holds for all sequences in* $[\![P]\!]$, $n = \mathrm{card}(E)$, *that* $T_1 \times T_2 ... \times T_n \subseteq \mathrm{Solutions}(\mathrm{tc}(\varepsilon^*))$.

Correctness of the operational semantics with respect to the event structures semantics

ε^*	Values of T_i for process $Score_{\varepsilon^*}$	$\mathrm{tc}(\varepsilon^*)$
$((i))$	$T_i \subseteq [1, n_\infty]$	$t_i \in \mathbb{N}$
$(i) \overset{\Delta}{\mapsto} (j)$	$T_i = \{1\} \wedge T_j \in T_i + \min(\Delta)$	$t_i \in \mathbb{N} \wedge t_j \in t_i$
$((i)) \overset{\Delta}{\mapsto} (j)$	$T_i \subseteq [1, n_\infty] \wedge T_j \in T_i + \min(\Delta)$	$t_i \in \mathbb{N} \wedge t_j \in t_i$
$((i)) \overset{\Delta}{\mapsto} ((j))$	$T_i \subseteq [1, n_\infty] \wedge T_j \in T_i + \Delta$	$t_i \in \mathbb{N} \wedge t_j \in t_i$
$(i) \overset{\Delta}{\mapsto} ((j))$	$T_i = \{1\} \wedge T_j \in T_i + \Delta$	$t_i \in \mathbb{N} \wedge t_j \in t_i$

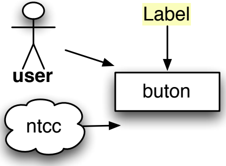
Table : Temporal constraints to launch the events in the ntcc model Vs. temporal constraints of the event structures semantics. We denote an interactive object as $((i))$ and a static one as (i).

Extension of Interactive Scores

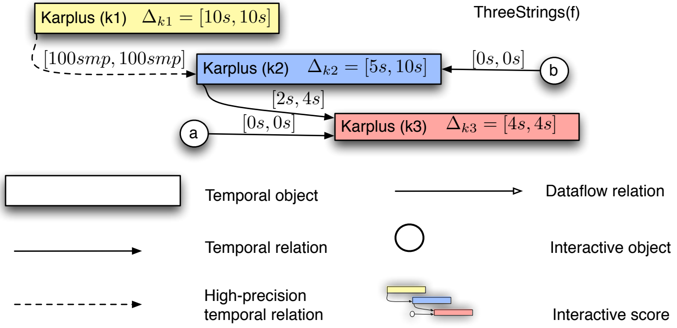


The novelty of our extension: Faust and ntcc interconnection

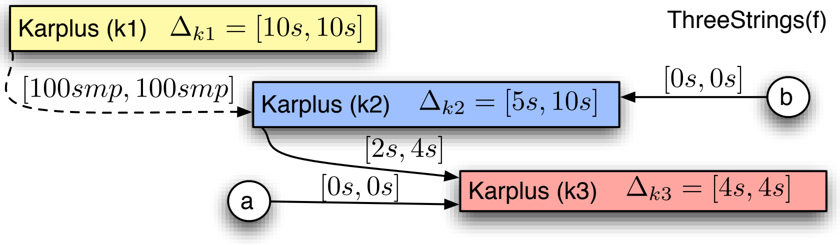
1. A model based on the *Non-deterministic Timed Concurrent Constraint* (ntcc) calculus for concurrency, user interactions and temporal relations, and
2. a model based on the *Faust* programming language for sound processing and micro controls.



Extension of Interactive Scores



Example: An arpeggio with three strings



Example: An arpeggio with three strings

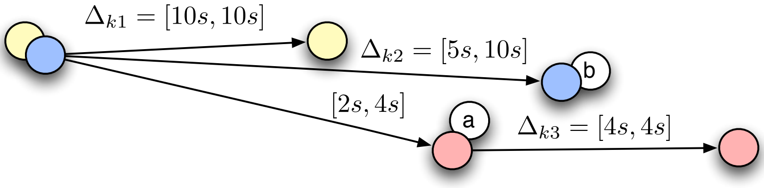


Figure : The constraint graph

Example: An arpeggio with three strings

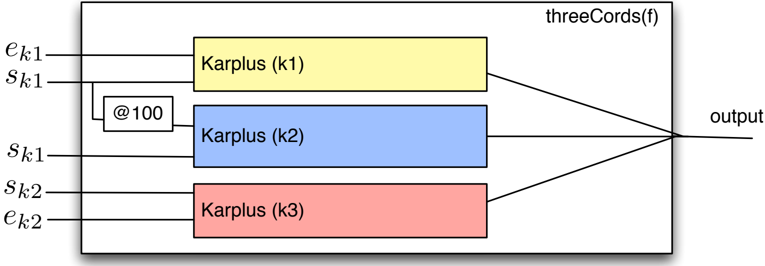


Figure : Faust's Block diagram

Example: An arpeggio with three strings

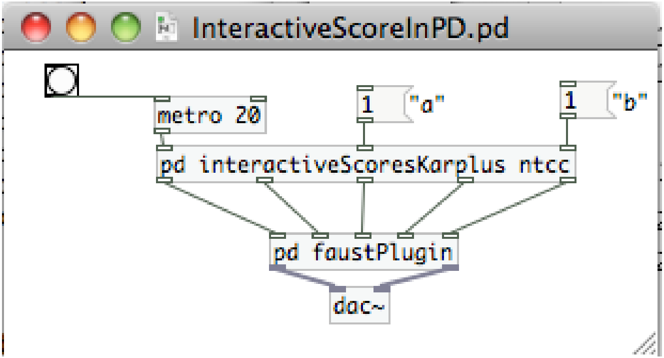


Figure : Implementation

Application: Three user controlled arpeggios

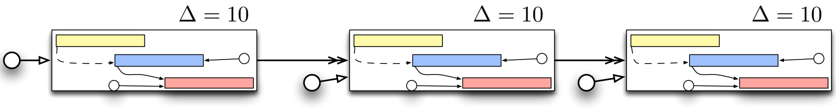
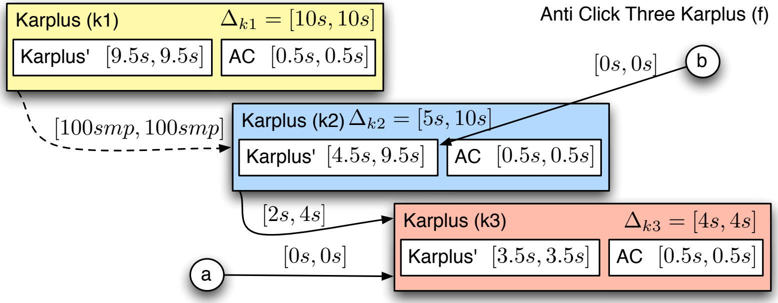


Figure : The double-headed arrow represents an inequality ($\leq$) and a white-headed arrow represents an equality relation ($=$).

Application: An arpeggio without “clicks”



Application: Sound source perception