Integrality of minimal unit circular-arc models

Francisco J. Soulignac
Pablo Terlisky

Universidad de Quilmes,
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

VII LAWCG
La Plata, Argentina
November 9, 2016

UCA, UIG, and PIG models

Definition (PCA model)

A PCA model is a finite family of inclusion-free arcs over a circle.

UCA, UIG, and PIG models

Definition (PCA model)

A PCA model is a finite family of inclusion-free arcs over a circle.

Definition (UCA model)

A UCA model is a PCA model in which all the arcs have the same length. A (c, ℓ)-CA model is one in which all arcs have length ℓ and, the circle has circumferece c and there is a distance of at least 1 between any two extremes of $\mathcal{M}$.

UCA, UIG, and PIG models

Definition (PCA model)

A PCA model is a finite family of inclusion-free arcs over a circle.

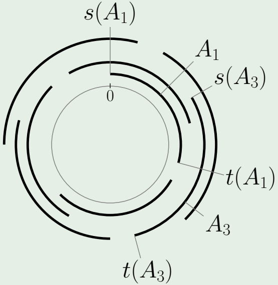
Definition (UCA model)

A UCA model is a PCA model in which all the arcs have the same length. A (c, ℓ)-CA model is one in which all arcs have length ℓ and, the circle has circumferece c and there is a distance of at least 1 between any two extremes of $\mathcal{M}$.

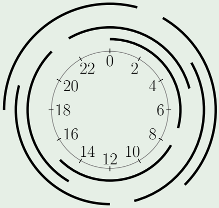
Definition (PIG and UIG models)

PIG and UIG models are those PCA and UCA models in which some point of the circle is crossed by no arcs, respectively. A ℓ-IG model is one in which all arcs have length ℓ.

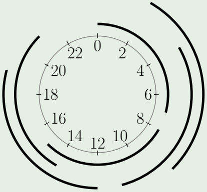
Example (PCA, (c, ℓ)-CA, ℓ-IG)



PCA model



$(24, 7)$-CA model



7-IG model

Basic properties

Definition (Model Equivalence)

Two PCA models $\mathcal{M}_1$, $\mathcal{M}_2$ are equivalent when the extremes of $\mathcal{M}_1$ appear in the same order as in $\mathcal{M}_2$ when their respective circles are transversed from 0.

Basic properties

Definition (Model Equivalence)

Two PCA models $\mathcal{M}_1$, $\mathcal{M}_2$ are equivalent when the extremes of $\mathcal{M}_1$ appear in the same order as in $\mathcal{M}_2$ when their respective circles are transversed from 0.

Definition (Minimality)

A (c, ℓ)-CA model $\mathcal{M}$ is minimal when $\ell \leq \ell'$ and $c \leq c'$ for every (c', ℓ')-CA model equivalent to $\mathcal{M}$.

Basic properties

Definition (Model Equivalence)

Two PCA models $\mathcal{M}_1$, $\mathcal{M}_2$ are equivalent when the extremes of $\mathcal{M}_1$ appear in the same order as in $\mathcal{M}_2$ when their respective circles are transversed from 0.

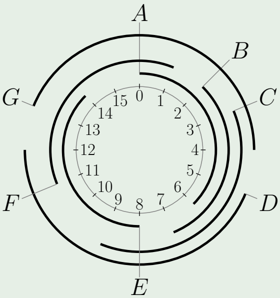
Definition (Minimality)

A (c, ℓ)-CA model $\mathcal{M}$ is minimal when $\ell \leq \ell'$ and $c \leq c'$ for every (c', ℓ')-CA model equivalent to $\mathcal{M}$.

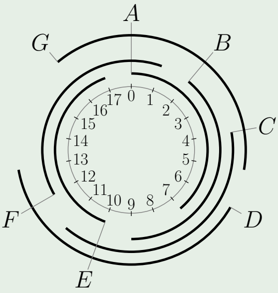
Definition (Strong Minimality)

A (c, ℓ)-CA model $\mathcal{M}$ with arcs $A_1, \ldots, A_n$ is strongly minimal when it is minimal and for any equivalent model $\mathcal{M}'$ with arcs $A'_1, \ldots, A'_n$, $s(A_i) \leq s(A'_i)$ for all $i \in \{1, \ldots, n\}$.

Example (PCA Model and its equivalent minimal model)



PCA model



$(18, 7)$-CA model (minimal)

What we knew about minimality

- All PIG models have an equivalent ℓ-CA model which is strongly minimal, for an integer ℓ (Pirlot, 1990).

What we knew about minimality

- All PIG models have an equivalent ℓ-CA model which is strongly minimal, for an integer ℓ (Pirlot, 1990).
- All UCA models have an equivalent (c, ℓ)-CA model which is minimal (Soulignac, 2014).

What we knew about minimality

- All PIG models have an equivalent ℓ-CA model which is strongly minimal, for an integer ℓ (Pirlot, 1990).
- All UCA models have an equivalent (c, ℓ)-CA model which is minimal (Soulignac, 2014).
- The integrality of these values for c and ℓ was conjectured but unknown.

The synthetic Graph of a model

Definition (Synthetic Graph)

The Synthetic Graph of a UCA model $\mathcal{M}$ is a multidigraph $\mathcal{S}$ which has a vertex $v(A)$ for each arc A of $\mathcal{M}$ and the following edges:

- $E_\sigma = \{v(A_i) \to v(A_{i+1}) \mid 1 \leq i < n\} \cup \{v(A_n) \to v(A_1)\}$
- $E_\nu = \{v(A_i) \to v(A_j) \mid t(A_i)s(A_j) \text{ are consecutive in } \mathcal{M}\}$
- $E_\eta = \{v(A_i) \to v(A_j) \mid s(A_i)t(A_j) \text{ are consecutive in } \mathcal{M}\}$

The elements of E_σ, E_ν and E_η are called the Steps, Noses and Hollows of $\mathcal{S}$, respectively.

The synthetic Graph of a model

Definition (Synthetic Graph)

The Synthetic Graph of a UCA model $\mathcal{M}$ is a multidigraph $\mathcal{S}$ which has a vertex $v(A)$ for each arc A of $\mathcal{M}$ and the following edges:

- $E_\sigma = \{v(A_i) \to v(A_{i+1}) \mid 1 \leq i < n\} \cup \{v(A_n) \to v(A_1)\}$
- $E_\nu = \{v(A_i) \to v(A_j) \mid t(A_i)s(A_j)$ are consecutive in $\mathcal{M}\}$
- $E_\eta = \{v(A_i) \to v(A_j) \mid s(A_i)t(A_j)$ are consecutive in $\mathcal{M}\}$

The elements of E_σ, E_ν and E_η are called the Steps, Noses and Hollows of $\mathcal{S}$, respectively.

- Observe that all equivalent PCA models share the same synthetic graph.

The synthetic Graph of a model

Definition (Synthetic Graph)

The Synthetic Graph of a UCA model $\mathcal{M}$ is a multidigraph $\mathcal{S}$ which has a vertex $v(A)$ for each arc A of $\mathcal{M}$ and the following edges:

- $E_\sigma = \{v(A_i) \to v(A_{i+1}) \mid 1 \leq i < n\} \cup \{v(A_n) \to v(A_1)\}$
- $E_\nu = \{v(A_i) \to v(A_j) \mid t(A_i)s(A_j) \text{ are consecutive in } \mathcal{M}\}$
- $E_\eta = \{v(A_i) \to v(A_j) \mid s(A_i)t(A_j) \text{ are consecutive in } \mathcal{M}\}$

The elements of E_σ, E_ν and E_η are called the Steps, Noses and Hollows of $\mathcal{S}$, respectively.

- Observe that all equivalent PCA models share the same synthetic graph.
- Jutta Mitas (1994) devised a way of representing synthetic graphs of PIG models which renders them as plane graphs.

Mitas' Drawing of a Synthetic Graph

- The vertices $\mathcal{S}$ are placed on a grid.

Mitas' Drawing of a Synthetic Graph

- The vertices $\mathcal{S}$ are placed on a grid.
- In the lowest row, the vertex $v(A_1)$ is placed in the leftmost position.

Mitas' Drawing of a Synthetic Graph

- The vertices $\mathcal{S}$ are placed on a grid.
- In the lowest row, the vertex $v(A_1)$ is placed in the leftmost position.
- Then all vertices of arcs that intersect A_1 are placed, from left to right, in the order the arcs appear on the model.

Mitas' Drawing of a Synthetic Graph

- The vertices $\mathcal{S}$ are placed on a grid.
- In the lowest row, the vertex $v(A_1)$ is placed in the leftmost position.
- Then all vertices of arcs that intersect A_1 are placed, from left to right, in the order the arcs appear on the model.
- The leftmost vertex of the second row is $v(A_q)$, where A_q is the first arc non-adjacent to A_1.

Mitas' Drawing of a Synthetic Graph

- The vertices $\mathcal{S}$ are placed on a grid.
- In the lowest row, the vertex $v(A_1)$ is placed in the leftmost position.
- Then all vertices of arcs that intersect A_1 are placed, from left to right, in the order the arcs appear on the model.
- The leftmost vertex of the second row is $v(A_q)$, where A_q is the first arc non-adjacent to A_1.
- The rest of the vertices of the second row are set in the same manner as in the first one.

Mitas' Drawing of a Synthetic Graph

- The vertices $\mathcal{S}$ are placed on a grid.
- In the lowest row, the vertex $v(A_1)$ is placed in the leftmost position.
- Then all vertices of arcs that intersect A_1 are placed, from left to right, in the order the arcs appear on the model.
- The leftmost vertex of the second row is $v(A_q)$, where A_q is the first arc non-adjacent to A_1.
- The rest of the vertices of the second row are set in the same manner as in the first one.
- And so on.

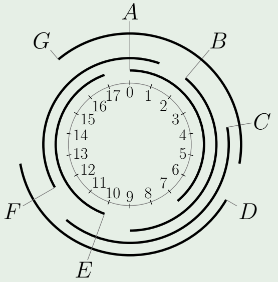
Backward edges

- A synthetic graph always has step edges that go from the rightmost vertex of a row to the leftmost vertex of the next. Also, it may have hollow edges that go from the rightmost vertex of a row to the leftmost vertex of the same row. We call these edges backward.

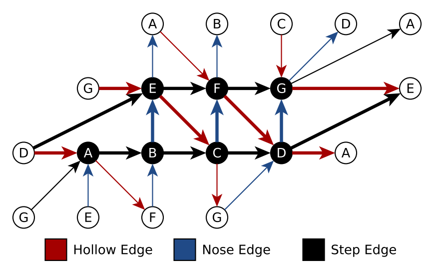
Backward edges

- A synthetic graph always has step edges that go from the rightmost vertex of a row to the leftmost vertex of the next. Also, it may have hollow edges that go from the rightmost vertex of a row to the leftmost vertex of the same row. We call these edges backward.
- The columns of the the vertices in each row are such that all non-backward edges point straight up or to the right.

Example (Synthetic Graph)



(18, 7)-CA model $\mathcal{M}$



Synthetic Graph $\mathcal{S}(\mathcal{M})$

Mitas' drawing was defined for PIG models, which didn't include edges that crossed 0. We call these edges external.

The weighting function: sep

Definition (sep function)

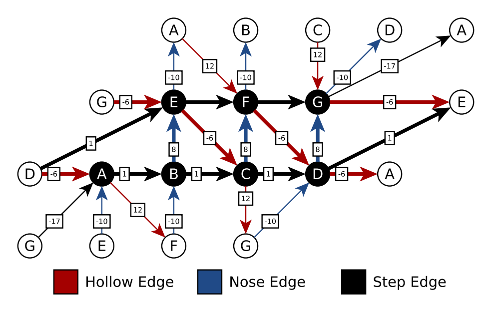
The sep function is a weighting function defined as:

- $\text{sep}(A_i \to A_j) = 1 - cq$ if $A_i \to A_j$ is a step
- $\text{sep}(A_i \to A_j) = 1 + \ell - cq$ if $A_i \to A_j$ is a nose
- $\text{sep}(A_i \to A_j) = 1 - \ell + cq$ if $A_i \to A_j$ is a hollow

where $q \in \{0, 1\}$ equals 1 if and only if A is external.

This function describes how far or close $s(A_i)$ and $s(A_j)$ must be in any (c, ℓ)-CA model equivalent to $\mathcal{M}$.

Example (Weighted Synthetic Graph)



Weighted Synthetic Graph of the previous example

sep value for cycles

For any walk $\mathcal{W}$ of a synthetic graph $\mathcal{S}$, we arrive a that:

$$\text{sep}(\mathcal{W}) = \ell\, \text{jmp}(\mathcal{W}) + c\, \text{ext}(\mathcal{W}) + |\mathcal{W}|$$

Where:

- jmp is the difference between the Noses and the Hollows of $\mathcal{W}$.
- ext is the difference between the external Hollows and the external Noses and steps of $\mathcal{W}$.

The key properties of synthetic graphs

- A PCA model $\mathcal{M}$ is equivalent to a (c, ℓ)-CA model if and only if $\text{sep}(\mathcal{W}) \leq 0$ for every cycle $\mathcal{W}$ fo $\mathcal{S}$.

The key properties of synthetic graphs

- A PCA model $\mathcal{M}$ is equivalent to a (c, ℓ)-CA model if and only if $\text{sep}(\mathcal{W}) \leq 0$ for every cycle $\mathcal{W}$ fo $\mathcal{S}$.
- Any internal cycle $\mathcal{W}$ includes exactly one backward edge, meaning that $\text{jmp}(\mathcal{W}) = -1$.

The key properties of synthetic graphs

- A PCA model $\mathcal{M}$ is equivalent to a (c, ℓ)-CA model if and only if $\text{sep}(\mathcal{W}) \leq 0$ for every cycle $\mathcal{W}$ fo $\mathcal{S}$.
- Any internal cycle $\mathcal{W}$ includes exactly one backward edge, meaning that $\text{jmp}(\mathcal{W}) = -1$.
- Since in PIG models all cycles are internal, one can solve ℓ by applying a maximum path algorithm less than n times.

The key properties of synthetic graphs

- A PCA model $\mathcal{M}$ is equivalent to a (c, ℓ)-CA model if and only if $\text{sep}(\mathcal{W}) \leq 0$ for every cycle $\mathcal{W}$ fo $\mathcal{S}$.
- Any internal cycle $\mathcal{W}$ includes exactly one backward edge, meaning that $\text{jmp}(\mathcal{W}) = -1$.
- Since in PIG models all cycles are internal, one can solve ℓ by applying a maximum path algorithm less than n times.
- However, this is not so on PCA graphs, as a cycle could take any external edge.

The κ-unrolling of a model

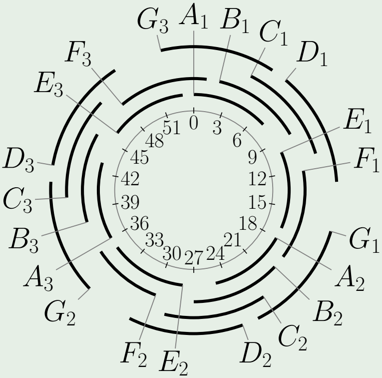
Definition (κ-unrolling)

The κ-unrolling of a PCA model $\mathcal{M}$ is the model with circumference κc that has κ arcs for each arc of $\mathcal{M}$ such that, for $0 \leq i < \kappa$:

$$s(A_i) = s(A) + ic, \text{ and } t(A_i) = t(A) + (i + q)c \bmod \kappa c,$$

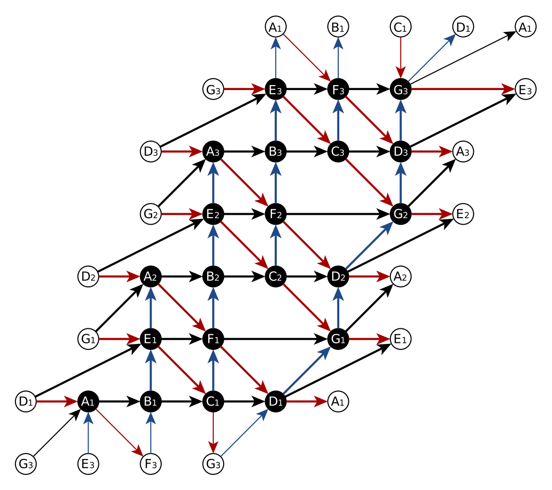
where $q \in \{0, 1\}$ equals 1 if and only if A is external.

Example (Unrolling)



3-unrolling $\kappa \cdot \mathcal{M}$ of the example model $\mathcal{M}$

Example (Unrolling - Synthetic Graph)



Synthetic graph of $\kappa \cdot \mathcal{M}$

Unrolling and cycles of $\mathcal{S}$

- The fundamental advantage of working with the unrolled version of a model is that circuits become paths.

Unrolling and cycles of $\mathcal{S}$

- The fundamental advantage of working with the unrolled version of a model is that circuits become paths.
- Specifically, circuits which have $\text{ext} = 0$ become internal circuits, while circuits with $\text{ext} \neq 0$ end at a different copy of the same vertex they started at, which is easy to predict.

Unrolling and cycles of $\mathcal{S}$

- The fundamental advantage of working with the unrolled version of a model is that circuits become paths.
- Specifically, circuits which have ext $= 0$ become internal circuits, while circuits with ext $\neq 0$ end at a different copy of the same vertex they started at, which is easy to predict.
- Therefore, for some "large enough" κ, the information of the external cycles of a model $\mathcal{M}$ is embedded into internal cycles of $\kappa \cdot \mathcal{M}$.

Finding ℓ and c by unrolling

- Clearly, any internal cycle of $\kappa \cdot \mathcal{M}$ is a cycle of $\mathcal{M}$.
- We'd like to show that at least one 'limiting cycle' of $\mathcal{M}$ is an internal cycle of $\kappa \cdot \mathcal{M}$.

Properties of cycles

The following results let us arrive at that conclusion:

Lemma

The synthetic graph of a minimal model $\mathcal{M}$ has circuits $\mathcal{W}_N$ and $\mathcal{W}_H$ with $\text{sep}(\mathcal{W}_N) = \text{sep}(\mathcal{W}_H) = 0$ such that $\text{ext}(\mathcal{W}_N) < 0$ and $\text{ext}(\mathcal{W}_H) \geq 0$.

Theorem (UCA $\Leftrightarrow$ Crossings)

A PCA model $\mathcal{M}$ is UCA if and only if all of its cycles with different signs of ext have a common vertex.

The previous results allow us to build a circuit W_0 with $\text{sep} = 0$ and $\text{ext} = 0$, meaning that the following of the edges of $\mathcal{W}_0$ in $\kappa \cdot \mathcal{M}$ is an internal cycle.

ℓ and c are Integers

Moreover, the following Lemma also holds:

Lemma

The synthetic graph of a minimal model $\mathcal{M}$ contains a circuit $\mathcal{W}_1$ with $\mathrm{sep} = 0$ *and* $\mathrm{ext} = -1$

Now, we can write the sep of $\mathcal{T}_0$ and $\mathcal{W}_1$ as:

$$0 = -\ell + |\mathcal{T}_0| \quad (1)$$

$$0 = \ell\,\mathrm{jmp}_1 - c + |\mathcal{W}_1| \quad (2)$$

From which can deduce that both ℓ and c are Integers.

An algorithm to find c and ℓ

We obtain ℓ by using the method for internal cycles on $\kappa \cdot \mathcal{M}$.
Fortunately, we have a bound for κ:

Lemma

For any circuit $\mathcal{W}$ of $\mathcal{S}$ with $\text{ext} = 0$, there is an internal circuit $\mathcal{W}'$ of $\kappa \cdot \mathcal{S}$ such that $\text{ext}(\mathcal{W}') = 0$ and $\text{sep}(\mathcal{W}') = \text{sep}(\mathcal{W})$, where $\kappa < 3n$.

However, with this bound we increase the cost of the solution by a factor of n, leaving the cost of finding ℓ at $O(n^3)$.
The value of c can then be found by binary search with cost $O(n^2 \lg n)$

New characterization algorithm

Let us state a more complete version of a previously stated theorem:

Theorem (UCA $\Leftrightarrow$ Crossings, complete)

The following are equivalent:

i) *The model $\mathcal{M}$ is UCA.*

ii) *Every pair of cycles of $\mathcal{M}$ with different signs of* ext *have a common vertex.*

iii) *Some greedy nose cycle and some greedy hollow cycle of $\mathcal{M}$ have a common vertex.*

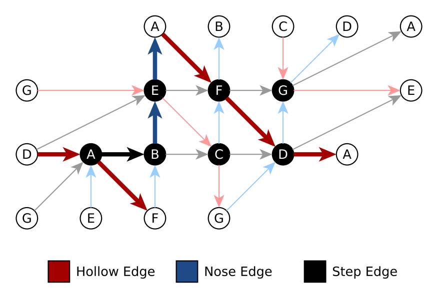
Where a greedy nose (hollow) cycle is a cycle $(v_1, \ldots, v_k)$ in which every edge $(v_i \to v_{i+1})$ is a nose (resp. hollow), or a step, only if there's no nose (resp. hollow) starting from v_i.

UCA recognition Algorithm

This theorem leaves us with a nice and simple recognition algorithm:

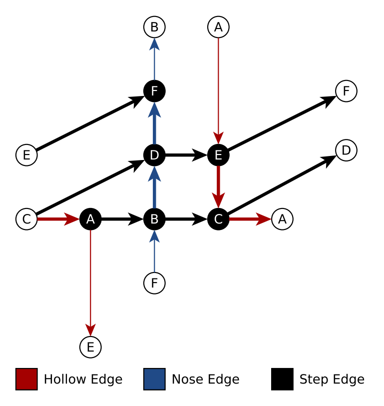
- Starting by any vertex, follow a nose edge, if available, or else a step. Continue iteratively until reaching a cycle.
- Do the same with hollow edges.
- The input model has a UCA equivalent if and only if the two found cycles have a vertex in common.

Example (UCA model)



Example showing the crossing between a greedy nose cycle and a greedy hollow cycle.

Example (non-UCA model)



Synthetic graph of a non-UCA model (Pyramid)

Conclusions

In this work we:

- Introduce the unrolling operation for PCA models,

Conclusions

In this work we:

- Introduce the unrolling operation for PCA models,
- Prove that the values of c and ℓ are integers for any UCA model,

Conclusions

In this work we:

- Introduce the unrolling operation for PCA models,
- Prove that the values of c and ℓ are integers for any UCA model,
- Show an algorithm to find these values in $O(n^3)$ time, and

Conclusions

In this work we:

- Introduce the unrolling operation for PCA models,
- Prove that the values of c and ℓ are integers for any UCA model,
- Show an algorithm to find these values in $O(n^3)$ time, and
- Show a new, simple algorithm to identify UCA models.

Minimal and minimum models

This work concerns *minimal models*. So, we concentrate on finding the values of c and ℓ that are minimum amongst the set of all *equivalent models*.

Minimal and minimum models

This work concerns *minimal models*. So, we concentrate on finding the values of c and ℓ that are minimum amongst the set of all *equivalent models*.

Besides searching for more efficient ways of finding these minimal values, we would like to have the minimum c and ℓ. These values would be those that are minimum amongst the set of all models that *represent the same circular-arc graph*, which may contain non-equivalent models.

Thank You.