

Constructive Category Theory and Tilting Equivalences via Strong Exceptional Sequences

Kamal Saleh

University of Siegen

March 09, 2022

1 Constructive Categories and Towers of Categories

Overview

1 Constructive Categories and Towers of Categories

2 Homomorphism Structures

- 1 Constructive Categories and Towers of Categories
- 2 Homomorphism Structures
- 3 Tilting Equivalences in Bounded Homotopy Categories

Section 1

Constructive Categories and Towers of Categories

Categories

Definition

A category \mathcal{C} consists of the following data:

Categories

Definition

A category \mathcal{C} consists of the following data:

- 1 Obj

 A B C

Categories

Definition

A category \mathcal{C} consists of the following data:

- 1 Obj
- 2 $\text{Hom}(A, B)$

$$A \longrightarrow B \quad C$$

Categories

Definition

A category \mathcal{C} consists of the following data:

- 1 Obj
- 2 $\text{Hom}(A, B)$

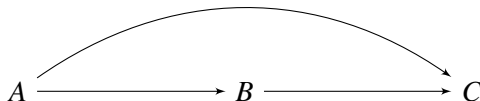
$$A \longrightarrow B \longrightarrow C$$

Categories

Definition

A category \mathcal{C} consists of the following data:

- 1 Obj
- 2 $\text{Hom}(A, B)$
- 3 $\cdot : \text{Hom}(A, B) \times \text{Hom}(B, C) \rightarrow \text{Hom}(A, C)$

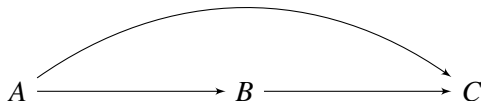


Categories

Definition

A category \mathcal{C} consists of the following data:

- 1 Obj
- 2 $\text{Hom}(A, B)$
- 3 $\cdot : \text{Hom}(A, B) \times \text{Hom}(B, C) \rightarrow \text{Hom}(A, C)$
- 4 Associative property

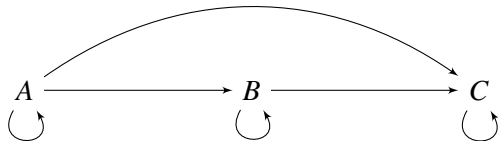


Categories

Definition

A category \mathcal{C} consists of the following data:

- 1 Obj
- 2 $\text{Hom}(A, B)$
- 3 $\cdot : \text{Hom}(A, B) \times \text{Hom}(B, C) \rightarrow \text{Hom}(A, C)$
- 4 Associative property
- 5 neutral element: $\text{id}_A \in \text{Hom}(A, A)$



Constructive categories

Constructive categories

A category becomes **constructive** through

Constructive categories

- A category becomes **constructive** through
- ▶ data structures for *objects* and *morphisms*,

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms and *identity morphisms* of objects.

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms and *identity morphisms* of objects.

A constructive category **has decidable equality of morphisms** through an algorithm which for A, B and $\alpha, \beta \in \text{Hom}(A, B)$ decides whether $\alpha = \beta$.

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms and *identity morphisms* of objects.

A constructive category **has decidable equality of morphisms** through an algorithm which for A, B and $\alpha, \beta \in \text{Hom}(A, B)$ decides whether $\alpha = \beta$.

Example: \mathbb{Q} -mat

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms and *identity morphisms* of objects.

A constructive category **has decidable equality of morphisms** through an algorithm which for A, B and $\alpha, \beta \in \text{Hom}(A, B)$ decides whether $\alpha = \beta$.

Example: \mathbb{Q} -mat

1

2

1

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms and *identity morphisms* of objects.

A constructive category **has decidable equality of morphisms** through an algorithm which for A, B and $\alpha, \beta \in \text{Hom}(A, B)$ decides whether $\alpha = \beta$.

Example: \mathbb{Q} -mat

1

2

1

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms and *identity morphisms* of objects.

A constructive category **has decidable equality of morphisms** through an algorithm which for A, B and $\alpha, \beta \in \text{Hom}(A, B)$ decides whether $\alpha = \beta$.

Example: \mathbb{Q} -mat

$$1 \xrightarrow{\begin{pmatrix} 1 & 2 \end{pmatrix}} 2 \xrightarrow{\begin{pmatrix} 3 \\ 4 \end{pmatrix}} 1$$

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms and *identity morphisms* of objects.

A constructive category **has decidable equality of morphisms** through an algorithm which for A, B and $\alpha, \beta \in \text{Hom}(A, B)$ decides whether $\alpha = \beta$.

Example: \mathbb{Q} -mat

$$1 \xrightarrow{\begin{pmatrix} 1 & 2 \end{pmatrix}} 2 \xrightarrow{\begin{pmatrix} 3 \\ 4 \end{pmatrix}} 1$$

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms and *identity morphisms* of objects.

A constructive category **has decidable equality of morphisms** through an algorithm which for A, B and $\alpha, \beta \in \text{Hom}(A, B)$ decides whether $\alpha = \beta$.

Example: \mathbb{Q} -mat

$$(1 \ 2) \cdot \begin{pmatrix} 3 \\ 4 \end{pmatrix} = (11)$$

$$1 \xrightarrow{(1 \ 2)} 2 \xrightarrow{\begin{pmatrix} 3 \\ 4 \end{pmatrix}} 1$$

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms and *identity morphisms* of objects.

A constructive category **has decidable equality of morphisms** through an algorithm which for A, B and $\alpha, \beta \in \text{Hom}(A, B)$ decides whether $\alpha = \beta$.

Example: \mathbb{Q} -mat

$$(1 \ 2) \cdot \begin{pmatrix} 3 \\ 4 \end{pmatrix} = (11)$$

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms and *identity morphisms* of objects.

A constructive category **has decidable equality of morphisms** through an algorithm which for A, B and $\alpha, \beta \in \text{Hom}(A, B)$ decides whether $\alpha = \beta$.

Example: \mathbb{Q} -mat

$$(1 \ 2) \cdot \begin{pmatrix} 3 \\ 4 \end{pmatrix} = (11)$$

$\begin{matrix} 1 & \xrightarrow{(1 \ 2)} & 2 & \xrightarrow{\begin{pmatrix} 3 \\ 4 \end{pmatrix}} & 1 \\ \textcircled{1} & & \textcircled{\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}} & & \textcircled{1} \end{matrix}$

Constructive categories

A category becomes **constructive** through

- ▶ data structures for *objects* and *morphisms*,
- ▶ algorithms to compute the *composition* of morphisms and *identity morphisms* of objects.

A constructive category **has decidable equality of morphisms** through an algorithm which for A, B and $\alpha, \beta \in \text{Hom}(A, B)$ decides whether $\alpha = \beta$.

Example: \mathbb{Q} -mat

$$(1 \ 2) \cdot \begin{pmatrix} 3 \\ 4 \end{pmatrix} = (11)$$

$$\begin{array}{ccccc} 1 & \xrightarrow{(1 \ 2)} & 2 & \xrightarrow{\begin{pmatrix} 3 \\ 4 \end{pmatrix}} & 1 \\ \textcircled{\uparrow} & & \textcircled{\uparrow} & & \textcircled{\uparrow} \\ (1) & & \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & & (1) \end{array}$$

Categorical Doctrines

- ▶ We use the term **categorical doctrine** to describe categories with specified additional properties or structures.

Categorical Doctrines

- ▶ We use the term **categorical doctrine** to describe categories with specified additional properties or structures.

Example: The doctrine of pre-Abelian Categories

The doctrine of pre-Abelian categories consists of categories in which

- 1 finite biproduct exists,
- 2 each morphism has an additive inverse,
- 3 kernels and cokernels exists.

Categorical Doctrines

- ▶ We use the term **categorical doctrine** to describe categories with specified additional properties or structures.

Example: The doctrine of pre-Abelian Categories

The doctrine of pre-Abelian categories consists of categories in which

- 1 finite biproduct exists,
- 2 each morphism has an additive inverse,
- 3 kernels and cokernels exists.

- ▶ A category \mathcal{P} is **constructive as an instance** of a doctrine \mathcal{D} if all disjunctions (\vee) and all existential quantifiers (\exists) in the defining axioms of the doctrine \mathcal{D} are realized in \mathcal{P} by algorithms.

Categorical Doctrines

- ▶ We use the term **categorical doctrine** to describe categories with specified additional properties or structures.

Example: The doctrine of pre-Abelian Categories

The doctrine of pre-Abelian categories consists of categories in which

- 1 finite biproduct exists,
- 2 each morphism has an additive inverse,
- 3 kernels and cokernels exists.

- ▶ A category \mathcal{P} is **constructive as an instance** of a doctrine \mathcal{D} if all disjunctions (\vee) and all existential quantifiers (\exists) in the defining axioms of the doctrine \mathcal{D} are realized in \mathcal{P} by algorithms.

The existential quantifiers of “kernels”

Let $\varphi \in \text{Hom}(A, B)$. The existence of a kernel to φ means

$$A \xrightarrow{\varphi} B$$

The existential quantifiers of “kernels”

Let $\varphi \in \text{Hom}(A, B)$. The existence of a kernel to φ means

... one have an object $\ker \varphi$,

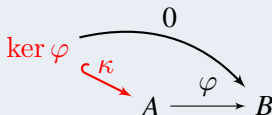
$\ker \varphi$

$$A \xrightarrow{\varphi} B$$

The existential quantifiers of “kernels”

Let $\varphi \in \text{Hom}(A, B)$. The existence of a kernel to φ means

... one have an object $\ker \varphi$,
its embedding $\kappa = \text{KernelEmbedding}(\varphi)$,



The existential quantifiers of “kernels”

Let $\varphi \in \text{Hom}(A, B)$. The existence of a kernel to φ means

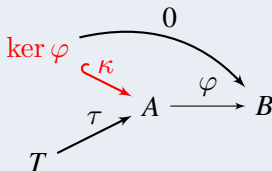
... one have an object $\ker \varphi$,
 its embedding $\kappa = \text{KernelEmbedding}(\varphi)$,
 and for every test morphism τ

$$\begin{array}{c}
 \ker \varphi \xrightarrow{\quad 0 \quad} B \\
 \searrow \kappa \quad \nearrow \varphi \\
 A \xrightarrow{\quad \quad} B \\
 T
 \end{array}$$

The existential quantifiers of “kernels”

Let $\varphi \in \text{Hom}(A, B)$. The existence of a kernel to φ means

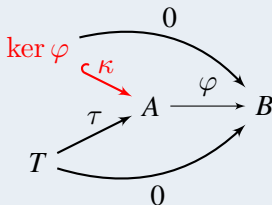
... one have an object $\ker \varphi$,
 its embedding $\kappa = \text{KernelEmbedding}(\varphi)$,
 and for every test morphism τ



The existential quantifiers of “kernels”

Let $\varphi \in \text{Hom}(A, B)$. The existence of a kernel to φ means

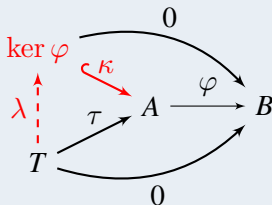
... one have an object $\ker \varphi$,
 its embedding $\kappa = \text{KernelEmbedding}(\varphi)$,
 and for every test morphism τ



The existential quantifiers of “kernels”

Let $\varphi \in \text{Hom}(A, B)$. The existence of a kernel to φ means

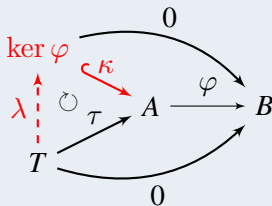
... one have an object $\ker \varphi$,
 its embedding $\kappa = \text{KernelEmbedding}(\varphi)$,
 and for every test morphism τ
 a unique morphism $\lambda = \text{KernelLift}(\varphi, \tau)$



The existential quantifiers of “kernels”

Let $\varphi \in \text{Hom}(A, B)$. The existence of a kernel to φ means

... one have an object $\ker \varphi$,
 its embedding $\kappa = \text{KernelEmbedding}(\varphi)$,
 and for every test morphism τ
 a unique morphism $\lambda = \text{KernelLift}(\varphi, \tau)$, such that



Category Constructors in CAP

CAP stands for *Categories, Algorithms and Programming*, is a software project for constructive category theory written in the computer algebra system GAP.

Category Constructors in CAP

CAP stands for *Categories, Algorithms and Programming*, is a software project for constructive category theory written in the computer algebra system GAP.

Two of the main design features of CAP are its support for:

Category Constructors in CAP

CAP stands for *Categories, Algorithms and Programming*, is a software project for constructive category theory written in the computer algebra system GAP.

Two of the main design features of CAP are its support for:

- ▶ defining *categorical doctrines*;

Category Constructors in CAP

CAP stands for *Categories, Algorithms and Programming*, is a software project for constructive category theory written in the computer algebra system GAP.

Two of the main design features of CAP are its support for:

- ▶ defining *categorical doctrines*;
- ▶ building *category constructors* which create instances of such doctrines.

Category Constructors in CAP

CAP stands for *Categories, Algorithms and Programming*, is a software project for constructive category theory written in the computer algebra system GAP.

Two of the main design features of CAP are its support for:

- ▶ defining *categorical doctrines*;
- ▶ building *category constructors* which create instances of such doctrines.

A **category constructor**, as the name suggests, is a procedure which outputs a category \mathcal{P} as an instance of a doctrine \mathbf{D} . The constructor

Category Constructors in CAP

CAP stands for *Categories, Algorithms and Programming*, is a software project for constructive category theory written in the computer algebra system GAP.

Two of the main design features of CAP are its support for:

- ▶ defining *categorical doctrines*;
- ▶ building *category constructors* which create instances of such doctrines.

A **category constructor**, as the name suggests, is a procedure which outputs a category \mathcal{P} as an instance of a doctrine \mathbf{D} . The constructor

- ▶ specifies the data structure of the objects and morphisms of the output category,

Category Constructors in CAP

CAP stands for *Categories, Algorithms and Programming*, is a software project for constructive category theory written in the computer algebra system GAP.

Two of the main design features of CAP are its support for:

- ▶ defining *categorical doctrines*;
- ▶ building *category constructors* which create instances of such doctrines.

A **category constructor**, as the name suggests, is a procedure which outputs a category \mathcal{P} as an instance of a doctrine \mathbf{D} . The constructor

- ▶ specifies the data structure of the objects and morphisms of the output category,
- ▶ expresses all defining categorical operations of the doctrine \mathbf{D} (of its output category) as algorithms written in terms of the provided input data.

Examples: An acyclic quiver q and a field k

Let k be a field and let q be an acyclic quiver.

 q k

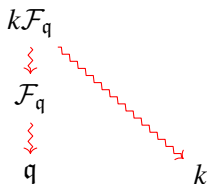
Examples: The free category over q

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc} \mathcal{F}_q & & \\ \Downarrow & & \\ q & & k \end{array}$$

Examples: The k -linear closure category

Let k be a field and let q be an acyclic quiver.



Examples: The quotient category by two-sided ideal

Let k be a field and let q be an acyclic quiver.

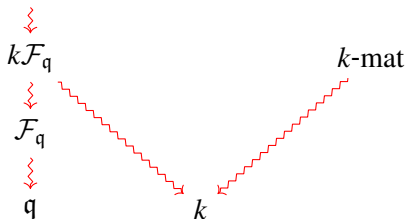
$$\mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle$$

ρ is a finite set in $\text{Mor}(k\mathcal{F}_q)$.

Examples: The category of matrices

Let k be a field and let q be an acyclic quiver.

$$\mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle$$



Examples: The f.d. ρ -bounded quiver representations

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 \Downarrow & \searrow & \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & k\text{-mat} \\
 \Downarrow & & \swarrow \\
 k\mathcal{F}_q & & \\
 \Downarrow & \searrow & \\
 \mathcal{F}_q & & k \\
 \Downarrow & & \swarrow \\
 q & &
 \end{array}$$

Examples: The category of bounded complexes

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \mathcal{C}^b(\text{rep}_k(q, \rho)) & & \\
 \Downarrow & & \\
 \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 \Downarrow & \searrow & \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & k\text{-mat} \\
 \Downarrow & & \swarrow \\
 k\mathcal{F}_q & & \\
 \Downarrow & \searrow & \swarrow \\
 \mathcal{F}_q & & k \\
 \Downarrow & & \\
 q & &
 \end{array}$$

Examples: The category of bounded complexes

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \mathcal{C}^b(\text{rep}_k(q, \rho)) & & \\
 \Downarrow & & \\
 \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 \Downarrow & \searrow & \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & k\text{-mat} \\
 \Downarrow & & \nearrow \\
 k\mathcal{F}_q & & \\
 \Downarrow & \searrow & \nearrow \\
 \mathcal{F}_q & & k \\
 \Downarrow & & \\
 q & & k
 \end{array}$$

They all have decidable equality of morphisms.

Examples: The bounded homotopy category?

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \mathcal{C}^b(\text{rep}_k(q, \rho)) & & \\
 \Downarrow & & \\
 \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 \Downarrow & \searrow & \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & k\text{-mat} \\
 \Downarrow & & \swarrow \\
 k\mathcal{F}_q & & \\
 \Downarrow & \searrow & \swarrow \\
 \mathcal{F}_q & & k \\
 \Downarrow & & \\
 q & & k
 \end{array}$$

What about the bounded homotopy category $\mathcal{K}^b(\text{rep}_k(q, \rho))$?

The bounded homotopy category

- ▶ Let \mathcal{P} be an additive category and let $\varphi: A \rightarrow B$ be in $\mathcal{C}^b(\mathcal{P})$.

The bounded homotopy category

- ▶ Let \mathcal{P} be an additive category and let $\varphi: A \rightarrow B$ be in $\mathcal{C}^b(\mathcal{P})$.

$$\cdots \longrightarrow A^{i-1} \xrightarrow{\partial_A^{i-1}} A^i \xrightarrow{\partial_A^i} A^{i+1} \longrightarrow \cdots$$

The bounded homotopy category

- Let \mathcal{P} be an additive category and let $\varphi: A \rightarrow B$ be in $\mathcal{C}^b(\mathcal{P})$.

$$\cdots \longrightarrow A^{i-1} \xrightarrow{\partial_A^{i-1}} A^i \xrightarrow{\partial_A^i} A^{i+1} \longrightarrow \cdots$$

$$\cdots \longrightarrow B^{i-1} \xrightarrow{\partial_B^{i-1}} B^i \xrightarrow{\partial_B^i} B^{i+1} \longrightarrow \cdots$$

The bounded homotopy category

- Let \mathcal{P} be an additive category and let $\varphi: A \rightarrow B$ be in $\mathcal{C}^b(\mathcal{P})$.

$$\begin{array}{ccccccc}
 \dots & \longrightarrow & A^{i-1} & \xrightarrow{\partial_A^{i-1}} & A^i & \xrightarrow{\partial_A^i} & A^{i+1} & \longrightarrow & \dots \\
 & & \downarrow \varphi^{i-1} & & \downarrow \varphi^i & & \downarrow \varphi^{i+1} & & \\
 \dots & \longrightarrow & B^{i-1} & \xrightarrow{\partial_B^{i-1}} & B^i & \xrightarrow{\partial_B^i} & B^{i+1} & \longrightarrow & \dots
 \end{array}$$

\circlearrowleft \circlearrowleft

The bounded homotopy category

- Let \mathcal{P} be an additive category and let $\varphi: A \rightarrow B$ be in $\mathcal{C}^b(\mathcal{P})$. Then φ is called **null-homotopic** if there exists a family of morphisms $(h^i: A^i \rightarrow B^{i-1})_{i \in \mathbb{Z}}$ such that $\partial_A^i \cdot h^{i+1} + h^i \cdot \partial_B^{i-1} = \varphi^i$ for all $i \in \mathbb{Z}$.

$$\begin{array}{ccccccc}
 \cdots & \longrightarrow & A^{i-1} & \xrightarrow{\partial_A^{i-1}} & A^i & \xrightarrow{\partial_A^i} & A^{i+1} & \longrightarrow & \cdots \\
 & & \downarrow \varphi^{i-1} & & \swarrow h^i & \downarrow \varphi^i & \nwarrow h^{i+1} & & \\
 \cdots & \longrightarrow & B^{i-1} & \xrightarrow{\partial_B^{i-1}} & B^i & \xrightarrow{\partial_B^i} & B^{i+1} & \longrightarrow & \cdots \\
 & & & & \downarrow \varphi^i & & \downarrow \varphi^{i+1} & &
 \end{array}$$

The bounded homotopy category

- Let \mathcal{P} be an additive category and let $\varphi: A \rightarrow B$ be in $\mathcal{C}^b(\mathcal{P})$. Then φ is called **null-homotopic** if there exists a family of morphisms $(h^i: A^i \rightarrow B^{i-1})_{i \in \mathbb{Z}}$ such that $\partial_A^i \cdot h^{i+1} + h^i \cdot \partial_B^{i-1} = \varphi^i$ for all $i \in \mathbb{Z}$.

$$\begin{array}{ccccccc}
 \cdots & \longrightarrow & A^{i-1} & \xrightarrow{\partial_A^{i-1}} & A^i & \xrightarrow{\partial_A^i} & A^{i+1} & \longrightarrow & \cdots \\
 & & \downarrow \varphi^{i-1} & & \swarrow h^i & \downarrow \varphi^i & \nwarrow h^{i+1} & & \\
 \cdots & \longrightarrow & B^{i-1} & \xrightarrow{\partial_B^{i-1}} & B^i & \xrightarrow{\partial_B^i} & B^{i+1} & \longrightarrow & \cdots \\
 & & & & \downarrow \varphi^i & & \downarrow \varphi^{i+1} & &
 \end{array}$$

- The **homotopy category**, denoted by $\mathcal{K}^b(\mathcal{P})$, is the quotient of $\mathcal{C}^b(\mathcal{P})$ by the (two sided) ideal of null-homotopic morphisms.

The bounded homotopy category

- Let \mathcal{P} be an additive category and let $\varphi: A \rightarrow B$ be in $\mathcal{C}^b(\mathcal{P})$. Then φ is called **null-homotopic** if there exists a family of morphisms $(h^i: A^i \rightarrow B^{i-1})_{i \in \mathbb{Z}}$ such that $\partial_A^i \cdot h^{i+1} + h^i \cdot \partial_B^{i-1} = \varphi^i$ for all $i \in \mathbb{Z}$.

$$\begin{array}{ccccccc}
 \cdots & \longrightarrow & A^{i-1} & \xrightarrow{\partial_A^{i-1}} & A^i & \xrightarrow{\partial_A^i} & A^{i+1} & \longrightarrow & \cdots \\
 & & \downarrow \varphi^{i-1} & & \swarrow h^i & \downarrow \varphi^i & \nwarrow h^{i+1} & & \\
 \cdots & \longrightarrow & B^{i-1} & \xrightarrow{\partial_B^{i-1}} & B^i & \xrightarrow{\partial_B^i} & B^{i+1} & \longrightarrow & \cdots \\
 & & & & \downarrow \varphi^i & & \downarrow \varphi^{i+1} & &
 \end{array}$$

- The **homotopy category**, denoted by $\mathcal{K}^b(\mathcal{P})$, is the quotient of $\mathcal{C}^b(\mathcal{P})$ by the (two sided) ideal of null-homotopic morphisms.
- That is, two morphisms $\varphi, \psi: A \rightarrow B$ in $\mathcal{K}^b(\mathcal{P})$ are equal if and only if there exists a solution to the linear system

$$\partial_A^i \cdot \chi^{i+1} + \chi^i \cdot \partial_B^{i-1} = \varphi^i - \psi^i$$

where i varies over a finite set of values.

Section 2

Homomorphism Structures

Homomorphism structures

Let \mathcal{P} and \mathcal{D} be categories.

Homomorphism structures

Let \mathcal{P} and \mathcal{D} be categories. A **\mathcal{D} -homomorphism structure** for \mathcal{P} consists of the following data:

Homomorphism structures

Let \mathcal{P} and \mathcal{D} be categories. A **\mathcal{D} -homomorphism structure** for \mathcal{P} consists of the following data:

- ▶ A distinguished object $1 \in \mathcal{D}$

Homomorphism structures

Let \mathcal{P} and \mathcal{D} be categories. A **\mathcal{D} -homomorphism structure** for \mathcal{P} consists of the following data:

- ▶ A distinguished object $1 \in \mathcal{D}$
- ▶ A functor $H(-, -): \mathcal{P}^{\text{op}} \times \mathcal{P} \rightarrow \mathcal{D}$

Homomorphism structures

Let \mathcal{P} and \mathcal{D} be categories. A **\mathcal{D} -homomorphism structure** for \mathcal{P} consists of the following data:

- ▶ A distinguished object $1 \in \mathcal{D}$
- ▶ A functor $H(-, -): \mathcal{P}^{\text{op}} \times \mathcal{P} \rightarrow \mathcal{D}$
- ▶ An isomorphism $\nu: \text{Hom}_{\mathcal{P}}(B, C) \cong \text{Hom}_{\mathcal{D}}(1, H(B, C))$ natural in B, C ,

Homomorphism structures

Let \mathcal{P} and \mathcal{D} be categories. A **\mathcal{D} -homomorphism structure** for \mathcal{P} consists of the following data:

- ▶ A distinguished object $1 \in \mathcal{D}$
- ▶ A functor $H(-, -): \mathcal{P}^{\text{op}} \times \mathcal{P} \rightarrow \mathcal{D}$
- ▶ An isomorphism $\nu: \text{Hom}_{\mathcal{P}}(B, C) \cong \text{Hom}_{\mathcal{D}}(1, H(B, C))$ natural in B, C ,

Homomorphism structures

Let \mathcal{P} and \mathcal{D} be categories. A **\mathcal{D} -homomorphism structure** for \mathcal{P} consists of the following data:

- ▶ A distinguished object $1 \in \mathcal{D}$
- ▶ A functor $H(-, -): \mathcal{P}^{\text{op}} \times \mathcal{P} \rightarrow \mathcal{D}$
- ▶ An isomorphism $\nu: \text{Hom}_{\mathcal{P}}(B, C) \cong \text{Hom}_{\mathcal{D}}(1, H(B, C))$ natural in B, C , i.e.,

$$\nu(\alpha \cdot \chi \cdot \beta) = \nu(\chi) \cdot H(\alpha, \beta)$$

for all compatible triples α, χ, β .

Example

- ▶ Let k be a field.

Example

- ▶ Let k be a field.
- ▶ $\mathcal{P} = \mathcal{D} = k\text{-mat}$

Example

- ▶ Let k be a field.
- ▶ $\mathcal{P} = \mathcal{D} = k\text{-mat}$
- ▶ The distinguished object is $1 \in k\text{-mat}$

Example

- ▶ Let k be a field.
- ▶ $\mathcal{P} = \mathcal{D} = k\text{-mat}$
- ▶ The distinguished object is $1 \in k\text{-mat}$
- ▶ H on objects: $H(m, n) = mn$

Example

- ▶ Let k be a field.
- ▶ $\mathcal{P} = \mathcal{D} = k\text{-mat}$
- ▶ The distinguished object is $1 \in k\text{-mat}$
- ▶ H on objects: $H(m, n) = mn$
- ▶ H on morphisms: $H(A, B) = A^{\text{tr}} \otimes B$

Example

- ▶ Let k be a field.
- ▶ $\mathcal{P} = \mathcal{D} = k\text{-mat}$
- ▶ The distinguished object is $1 \in k\text{-mat}$
- ▶ H on objects: $H(m, n) = mn$
- ▶ H on morphisms: $H(A, B) = A^{\text{tr}} \otimes B$
- ▶ For a morphism $A : m \rightarrow n$ we define $\nu(A) : 1 \rightarrow mn$ by the vectorization $\text{vec}(A)$ of A , i.e., the one-row matrix defined by the concatenation of all rows of A .

Example

- ▶ Let k be a field.
- ▶ $\mathcal{P} = \mathcal{D} = k\text{-mat}$
- ▶ The distinguished object is $1 \in k\text{-mat}$
- ▶ H on objects: $H(m, n) = mn$
- ▶ H on morphisms: $H(A, B) = A^{\text{tr}} \otimes B$
- ▶ For a morphism $A : m \rightarrow n$ we define $\nu(A) : 1 \rightarrow mn$ by the vectorization $\text{vec}(A)$ of A , i.e., the one-row matrix defined by the concatenation of all rows of A .
- ▶ The induced equality:

$$\text{vec}(M \cdot U \cdot N) = \text{vec}(U) \cdot (M^T \otimes N)$$

is known as the Kronecker product trick for solving two-sided matrix equations.

Two-sided linear systems

Two-sided linear systems

- Let \mathcal{P} be an additive category. A **linear system** in \mathcal{P} is a collection of morphisms α_{ij} , β_{ij} and γ_i of the following form:

$$\begin{array}{ccccccc}
 \alpha_{11} \cdot X_1 \cdot \beta_{11} & + \cdots + & \alpha_{1n} \cdot X_n \cdot \beta_{1n} & = & \gamma_1 \\
 \vdots & & \vdots & & \vdots \\
 \alpha_{m1} \cdot X_1 \cdot \beta_{m1} & + \cdots + & \alpha_{mn} \cdot X_n \cdot \beta_{mn} & = & \gamma_m
 \end{array}$$

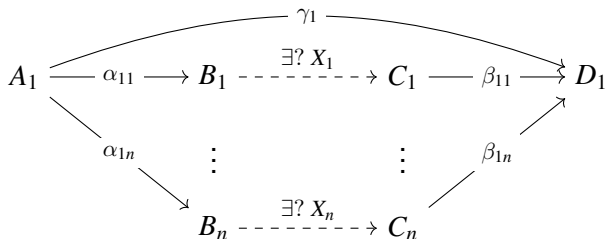
where the X_j are unknown morphisms.

Two-sided linear systems

- Let \mathcal{P} be an additive category. A **linear system** in \mathcal{P} is a collection of morphisms α_{ij} , β_{ij} and γ_i of the following form:

$$\begin{array}{ccccccc}
 \alpha_{11} \cdot X_1 \cdot \beta_{11} & + \cdots + & \alpha_{1n} \cdot X_n \cdot \beta_{1n} & = & \gamma_1 \\
 \vdots & & \vdots & & \vdots \\
 \alpha_{m1} \cdot X_1 \cdot \beta_{m1} & + \cdots + & \alpha_{mn} \cdot X_n \cdot \beta_{mn} & = & \gamma_m
 \end{array}$$

where the X_j are unknown morphisms.

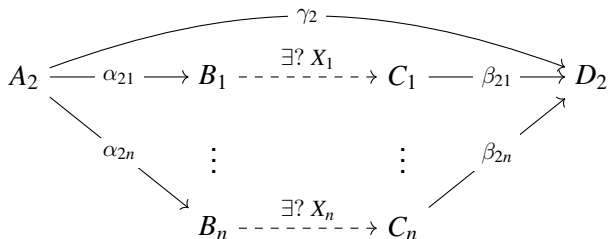


Two-sided linear systems

- Let \mathcal{P} be an additive category. A **linear system** in \mathcal{P} is a collection of morphisms α_{ij} , β_{ij} and γ_i of the following form:

$$\begin{array}{ccccccc}
 \alpha_{11} \cdot X_1 \cdot \beta_{11} & + \cdots + & \alpha_{1n} \cdot X_n \cdot \beta_{1n} & = & \gamma_1 \\
 \vdots & & \vdots & & \vdots \\
 \alpha_{m1} \cdot X_1 \cdot \beta_{m1} & + \cdots + & \alpha_{mn} \cdot X_n \cdot \beta_{mn} & = & \gamma_m
 \end{array}$$

where the X_j are unknown morphisms.

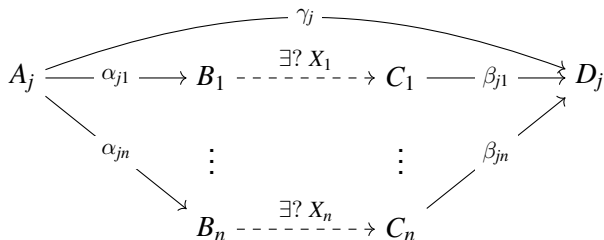


Two-sided linear systems

- Let \mathcal{P} be an additive category. A **linear system** in \mathcal{P} is a collection of morphisms α_{ij} , β_{ij} and γ_i of the following form:

$$\begin{array}{ccccccc} \alpha_{11} \cdot X_1 \cdot \beta_{11} & + \cdots + & \alpha_{1n} \cdot X_n \cdot \beta_{1n} & = & \gamma_1 \\ \vdots & & \vdots & & \vdots \\ \alpha_{m1} \cdot X_1 \cdot \beta_{m1} & + \cdots + & \alpha_{mn} \cdot X_n \cdot \beta_{mn} & = & \gamma_m \end{array}$$

where the X_j are unknown morphisms.

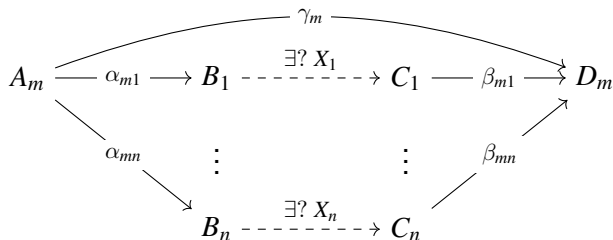


Two-sided linear systems

- Let \mathcal{P} be an additive category. A **linear system** in \mathcal{P} is a collection of morphisms α_{ij} , β_{ij} and γ_i of the following form:

$$\begin{array}{ccccccc}
 \alpha_{11} \cdot X_1 \cdot \beta_{11} & + \cdots + & \alpha_{1n} \cdot X_n \cdot \beta_{1n} & = & \gamma_1 \\
 \vdots & & \vdots & & \vdots \\
 \alpha_{m1} \cdot X_1 \cdot \beta_{m1} & + \cdots + & \alpha_{mn} \cdot X_n \cdot \beta_{mn} & = & \gamma_m
 \end{array}$$

where the X_j are unknown morphisms.



Two-sided linear systems

- ▶ Let \mathcal{D} be an additive category.

Two-sided linear systems

- ▶ Let \mathcal{D} be an additive category.
- ▶ Assume \mathcal{P} is equipped with a \mathcal{D} -homomorphism structure.

Two-sided linear systems

- ▶ Let \mathcal{D} be an additive category.
- ▶ Assume \mathcal{P} is equipped with a \mathcal{D} -homomorphism structure.
- ▶ There exists a solution in \mathcal{P} for the previous linear system iff there exists a solution in \mathcal{D} to the lift problem:

$$\begin{array}{ccc}
 & & 1 \\
 & \swarrow \text{\scriptsize } \exists? & \downarrow \text{\scriptsize } (\nu(\gamma_i))_{1i} \\
 \bigoplus_{j=1}^n H(B_j, C_j) & \xrightarrow{\text{\scriptsize } (H(\alpha_{ij}, \beta_{ij}))_{ji}} & \bigoplus_{i=1}^m H(A_i, D_i) \\
 & \circlearrowleft &
 \end{array}$$

Two-sided linear systems

- ▶ Let \mathcal{D} be an additive category.
- ▶ Assume \mathcal{P} is equipped with a \mathcal{D} -homomorphism structure.
- ▶ There exists a solution in \mathcal{P} for the previous linear system iff there exists a solution in \mathcal{D} to the lift problem:

$$\begin{array}{ccc}
 & & 1 \\
 & \swarrow \text{---} & \downarrow (\nu(\gamma_i))_{1i} \\
 & (\nu(X_j))_{1j} & \circlearrowleft \\
 \bigoplus_{j=1}^n H(B_j, C_j) & \xrightarrow{(H(\alpha_{ij}, \beta_{ij}))_{ji}} & \bigoplus_{i=1}^m H(A_i, D_i)
 \end{array}$$

Two-sided linear systems

- ▶ Let \mathcal{D} be an additive category.
- ▶ Assume \mathcal{P} is equipped with a \mathcal{D} -homomorphism structure.
- ▶ There exists a solution in \mathcal{P} for the previous linear system iff there exists a solution in \mathcal{D} to the lift problem:

$$\begin{array}{ccc}
 & & 1 \\
 & \swarrow^{(\nu(X_j))_{1j}} & \downarrow^{(\nu(\gamma_i))_{1i}} \\
 \bigoplus_{j=1}^n H(B_j, C_j) & \xrightarrow{(H(\alpha_{ij}, \beta_{ij}))_{ji}} & \bigoplus_{i=1}^m H(A_i, D_i)
 \end{array}$$

- ▶ That is, we can transfer a two-sided linear system in \mathcal{P} to a one-sided linear equation in \mathcal{D} .

Equality of morphisms in homotopy categories

Theorem

Let \mathcal{P} be an additive category equipped with a \mathcal{D} -homomorphism structure $(1, H(-, -), \nu)$ such that \mathcal{D} has decidable lifts, i.e., we can decide in \mathcal{D} the solvability of the one-sided equation $\mathcal{Y} \cdot \beta = \gamma$. Then

Equality of morphisms in homotopy categories

Theorem

Let \mathcal{P} be an additive category equipped with a \mathcal{D} -homomorphism structure $(1, H(-, -), \nu)$ such that \mathcal{D} has decidable lifts, i.e., we can decide in \mathcal{D} the solvability of the one-sided equation $\mathcal{Y} \cdot \beta = \gamma$. Then

- 1 $\mathcal{K}^b(\mathcal{P})$ has decidable equality of morphisms,

Equality of morphisms in homotopy categories

Theorem

Let \mathcal{P} be an additive category equipped with a \mathcal{D} -homomorphism structure $(1, H(-, -), \nu)$ such that \mathcal{D} has decidable lifts, i.e., we can decide in \mathcal{D} the solvability of the one-sided equation $\mathcal{Y} \cdot \beta = \gamma$. Then

- 1 $\mathcal{K}^b(\mathcal{P})$ has decidable equality of morphisms,
- 2 If \mathcal{D} is Abelian and $1 \in \mathcal{D}$ is a projective object, then $\mathcal{C}^b(\mathcal{P})$ and $\mathcal{K}^b(\mathcal{P})$ can be equipped with a \mathcal{D} -homomorphism structure.

Example: An acyclic quiver q and a field k

Let k be a field and let q be an acyclic quiver.

 q k

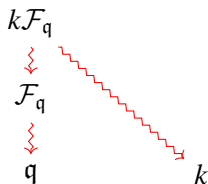
Example: The free category over q

Let k be a field and let q be an acyclic quiver.

 \mathcal{F}_q  q k

Example: The k -linear closure category

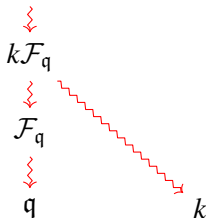
Let k be a field and let q be an acyclic quiver.



Example: The quotient category by two-sided ideal

Let k be a field and let q be an acyclic quiver.

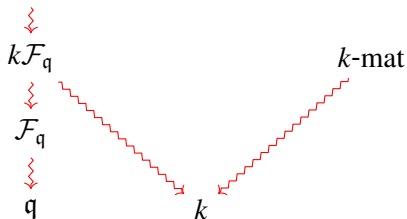
$$\mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle$$



Example: The category of matrices

Let k be a field and let q be an acyclic quiver.

$$\mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle$$



Example: The f.d. ρ -bounded quiver representations

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 \Downarrow & \searrow & \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & k\text{-mat} \\
 \Downarrow & & \swarrow \\
 k\mathcal{F}_q & & \\
 \Downarrow & \searrow & \\
 \mathcal{F}_q & & k \\
 \Downarrow & & \swarrow \\
 q & &
 \end{array}$$

The diagram illustrates the relationships between various mathematical objects. The top node is $\text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}]$. A vertical red zigzag arrow points down to $\mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle$. From \mathcal{A}_q , a vertical red zigzag arrow points down to $k\mathcal{F}_q$, and a red zigzag arrow points down-right to $k\text{-mat}$. From $k\mathcal{F}_q$, a vertical red zigzag arrow points down to \mathcal{F}_q , and a red zigzag arrow points down-right to k . From \mathcal{F}_q , a vertical red zigzag arrow points down to q . From k , a red zigzag arrow points up-right to $k\text{-mat}$.

Example: The category of bounded complexes

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \mathcal{C}^b(\text{rep}_k(q, \rho)) & & \\
 \Downarrow & & \\
 \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 \Downarrow & \searrow & \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & k\text{-mat} \\
 \Downarrow & & \swarrow \\
 k\mathcal{F}_q & & \\
 \Downarrow & \searrow & \\
 \mathcal{F}_q & & k \\
 \Downarrow & & \swarrow \\
 q & &
 \end{array}$$

Example: The bounded homotopy category

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \mathcal{C}^b(\text{rep}_k(q, \rho)) & \xleftarrow{\text{red wavy}} & \mathcal{K}^b(\text{rep}_k(q, \rho)) \\
 \downarrow \text{red wavy} & & \\
 \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 \downarrow \text{red wavy} & \searrow \text{red zigzag} & \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & k\text{-mat} \\
 \downarrow \text{red wavy} & & \\
 k\mathcal{F}_q & & \\
 \downarrow \text{red wavy} & \searrow \text{red zigzag} & \\
 \mathcal{F}_q & & k \\
 \downarrow \text{red wavy} & & \\
 q & &
 \end{array}$$

Example: The bounded homotopy category

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \mathcal{C}^b(\text{rep}_k(q, \rho)) & \xleftarrow{\text{red wavy}} & \mathcal{K}^b(\text{rep}_k(q, \rho)) \\
 \downarrow \text{red wavy} & & \\
 \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 \downarrow \text{red wavy} & \searrow \text{red zigzag} & \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & k\text{-mat} \\
 \downarrow \text{red wavy} & & \\
 k\mathcal{F}_q & & \\
 \downarrow \text{red wavy} & \searrow \text{red zigzag} & \\
 \mathcal{F}_q & & k \\
 \downarrow \text{red wavy} & & \\
 q & & k
 \end{array}$$

They all have decidable equality of morphisms.

Example: The bounded homotopy category

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \mathcal{C}^b(\text{rep}_k(q, \rho)) & \xleftarrow{\text{red wavy}} & \mathcal{K}^b(\text{rep}_k(q, \rho)) \\
 \downarrow \text{red wavy} & & \\
 \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 \downarrow \text{red wavy} & \searrow \text{red zigzag} & \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & k\text{-mat} \\
 \downarrow \text{red wavy} & & \\
 k\mathcal{F}_q & & \\
 \downarrow \text{red wavy} & \searrow \text{red zigzag} & \\
 \mathcal{F}_q & & k \\
 \downarrow \text{red wavy} & & \\
 q & & k
 \end{array}$$

What about the bounded derived category $\mathcal{D}^b(\text{rep}_k(q, \rho))$?

Example: The subcategory of projective objects

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \mathcal{C}^b(\text{rep}_k(q, \rho)) & \xleftarrow{\sim} & \mathcal{K}^b(\text{rep}_k(q, \rho)) \\
 \downarrow & & \\
 \text{proj}(\text{rep}_k(q, \rho)) & \xrightarrow{\sim} & \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] \\
 \downarrow & & \searrow \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & k\text{-mat} \\
 \downarrow & & \swarrow \\
 k\mathcal{F}_q & & k \\
 \downarrow & \searrow & \swarrow \\
 \mathcal{F}_q & & k \\
 \downarrow & & \\
 q & & k
 \end{array}$$

What about the bounded derived category $\mathcal{D}^b(\text{rep}_k(q, \rho))$?

Example: The bounded homotopy category

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \mathcal{K}^b(\text{proj}(\text{rep}_k(q, \rho))) & & \mathcal{C}^b(\text{rep}_k(q, \rho)) \xleftarrow{\sim} \mathcal{K}^b(\text{rep}_k(q, \rho)) \\
 \Downarrow & & \Downarrow \\
 \text{proj}(\text{rep}_k(q, \rho)) \rightsquigarrow \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 & & \Downarrow \\
 & & \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle \\
 & & \Downarrow \\
 & & k\mathcal{F}_q \qquad \qquad \qquad k\text{-mat} \\
 & & \Downarrow \qquad \qquad \qquad \Downarrow \\
 & & \mathcal{F}_q \qquad \qquad \qquad k \\
 & & \Downarrow \qquad \qquad \qquad \Downarrow \\
 & & q \qquad \qquad \qquad k
 \end{array}$$

What about the bounded derived category $\mathcal{D}^b(\text{rep}_k(q, \rho))$?

Example: The bounded derived category

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \mathcal{K}^b(\text{proj}(\text{rep}_k(q, \rho))) & & \mathcal{C}^b(\text{rep}_k(q, \rho)) \xleftarrow{\sim} \mathcal{K}^b(\text{rep}_k(q, \rho)) \\
 \Downarrow & & \Downarrow \\
 \text{proj}(\text{rep}_k(q, \rho)) \rightsquigarrow \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 \Downarrow & & \Downarrow \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & \\
 \Downarrow & & \Downarrow \\
 k\mathcal{F}_q & & k\text{-mat} \\
 \Downarrow & & \Downarrow \\
 \mathcal{F}_q & & \\
 \Downarrow & & \Downarrow \\
 q & & k
 \end{array}$$

The diagram illustrates the relationships between various categories and their quotients. The top row shows the equivalence $\mathcal{C}^b(\text{rep}_k(q, \rho)) \xleftarrow{\sim} \mathcal{K}^b(\text{rep}_k(q, \rho))$. Below this, the projective part of the representation category is shown to be equivalent to the representation category itself: $\text{proj}(\text{rep}_k(q, \rho)) \rightsquigarrow \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}]$. The algebra \mathcal{A}_q is defined as the quotient $k\mathcal{F}_q / \langle \rho \rangle$. The diagram shows a series of downward arrows (some dashed, some solid) from \mathcal{A}_q to $k\mathcal{F}_q$, \mathcal{F}_q , and q . Similarly, there are downward arrows from $k\mathcal{F}_q$ to k and from $k\text{-mat}$ to k . A large dashed arrow also points from \mathcal{A}_q to k . A solid arrow points from $k\mathcal{F}_q$ to $k\text{-mat}$, and another solid arrow points from $k\text{-mat}$ to k .

If $\text{rep}_k(q, \rho)$ has a f.g. dimension then $\mathcal{D}^b(\text{rep}_k(q, \rho)) \simeq \mathcal{K}^b(\text{proj}(\text{rep}_k(q, \rho)))$.

Example: The bounded derived category

Let k be a field and let q be an acyclic quiver.

$$\begin{array}{ccc}
 \mathcal{K}^b(\text{proj}(\text{rep}_k(q, \rho))) & & \mathcal{C}^b(\text{rep}_k(q, \rho)) \xleftarrow{\sim} \mathcal{K}^b(\text{rep}_k(q, \rho)) \\
 \Downarrow & & \Downarrow \\
 \text{proj}(\text{rep}_k(q, \rho)) \xrightarrow{\sim} \text{rep}_k(q, \rho) := [\mathcal{A}_q, k\text{-mat}] & & \\
 \Downarrow & & \Downarrow \\
 \mathcal{A}_q := k\mathcal{F}_q / \langle \rho \rangle & & \\
 \Downarrow & & \Downarrow \\
 k\mathcal{F}_q & & k\text{-mat} \\
 \Downarrow & & \Downarrow \\
 \mathcal{F}_q & & \\
 \Downarrow & & \Downarrow \\
 q & & k
 \end{array}$$

The diagram illustrates the relationships between various categories and structures. The top row shows the equivalence between the bounded derived category of projective representations and the bounded derived category of representations. The middle row shows the equivalence between the projective representations and the representations, which are identified as the derived category of the quotient algebra \mathcal{A}_q and k -matrices. The bottom part of the diagram shows the relationships between the quotient algebra \mathcal{A}_q , the free algebra $k\mathcal{F}_q$, the free algebra \mathcal{F}_q , the quiver q , and the field k . The arrows indicate the natural maps and inclusions between these structures.

They all have decidable equality of morphisms.

Section 3

Tilting Equivalences in Bounded Homotopy Categories

Triangles

A **category with shift functor** is a pair (\mathfrak{T}, Σ) given by

Triangles

A **category with shift functor** is a pair (\mathcal{T}, Σ) given by

- ▶ An additive category \mathcal{T} .

Triangles

A **category with shift functor** is a pair (\mathcal{T}, Σ) given by

- ▶ An additive category \mathcal{T} .
- ▶ An autoequivalence $\Sigma : \mathcal{T} \rightarrow \mathcal{T}$ called **shift**.

Triangles

A **category with shift functor** is a pair (\mathcal{T}, Σ) given by

- ▶ An additive category \mathcal{T} .
- ▶ An autoequivalence $\Sigma : \mathcal{T} \rightarrow \mathcal{T}$ called **shift**.

A **triangle** over (\mathcal{T}, Σ) is a sequence

$$A \xrightarrow{\alpha} B \xrightarrow{\iota} C \xrightarrow{\pi} \Sigma A.$$

Triangles

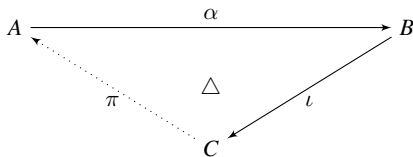
A **category with shift functor** is a pair (\mathcal{T}, Σ) given by

- ▶ An additive category \mathcal{T} .
- ▶ An autoequivalence $\Sigma : \mathcal{T} \rightarrow \mathcal{T}$ called **shift**.

A **triangle** over (\mathcal{T}, Σ) is a sequence

$$A \xrightarrow{\alpha} B \xrightarrow{\iota} C \xrightarrow{\pi} \Sigma A.$$

A triangle may be depicted by



Triangulated categories

A **triangulated category** is defined by the following data:

Triangulated categories

A **triangulated category** is defined by the following data:

- ▶ a category with shift functor (\mathcal{T}, Σ)

Triangulated categories

A **triangulated category** is defined by the following data:

- ▶ a category with shift functor (\mathcal{T}, Σ)
- ▶ distinguished class of triangles Δ , called the **exact triangles**

subject to many Axioms:

Triangulated categories

A **triangulated category** is defined by the following data:

- ▶ a category with shift functor (\mathcal{T}, Σ)
- ▶ distinguished class of triangles Δ , called the **exact triangles**

subject to many Axioms:

- ▶ Any morphism $\alpha: A \rightarrow B$

Triangulated categories

A **triangulated category** is defined by the following data:

- ▶ a category with shift functor (\mathfrak{T}, Σ)
- ▶ distinguished class of triangles Δ , called the **exact triangles**

subject to many Axioms:

- ▶ Any morphism $\alpha: A \rightarrow B$ can be completed to an exact triangle

$$A \xrightarrow{\alpha} B \xrightarrow{\iota(\alpha)} \text{Cone}(\alpha) \xrightarrow{\pi(\alpha)} \Sigma A \in \Delta.$$

Triangulated categories

A **triangulated category** is defined by the following data:

- ▶ a category with shift functor (\mathfrak{T}, Σ)
- ▶ distinguished class of triangles Δ , called the **exact triangles**

subject to many Axioms:

- ▶ Any morphism $\alpha: A \rightarrow B$ can be completed to an exact triangle

$$A \xrightarrow{\alpha} B \xrightarrow{\iota(\alpha)} \text{Cone}(\alpha) \xrightarrow{\pi(\alpha)} \Sigma A \in \Delta.$$

Triangulated categories

A **triangulated category** is defined by the following data:

- ▶ a category with shift functor (\mathfrak{T}, Σ)
- ▶ distinguished class of triangles Δ , called the **exact triangles**

subject to many Axioms:

- ▶ Any morphism $\alpha: A \rightarrow B$ can be completed to an exact triangle

$$A \xrightarrow{\alpha} B \xrightarrow{\iota(\alpha)} \text{Cone}(\alpha) \xrightarrow{\pi(\alpha)} \Sigma A \in \Delta.$$

- ▶ The rotation axiom, the octahedral axiom, etc ...

Example

(Bounded) homotopy and derived categories are triangulated.

Strong exceptional sequences

Let k be a field and let $(\mathcal{T}, \Sigma, \Delta)$ be a k -linear triangulated category.

Strong exceptional sequences

Let k be a field and let $(\mathcal{T}, \Sigma, \Delta)$ be a k -linear triangulated category.

- ▶ A **strong exceptional sequence** is a sequence of objects $\mathcal{E} = (E_1, \dots, E_m)$ in \mathcal{T} such that

Strong exceptional sequences

Let k be a field and let $(\mathcal{T}, \Sigma, \Delta)$ be a k -linear triangulated category.

- ▶ A **strong exceptional sequence** is a sequence of objects $\mathcal{E} = (E_1, \dots, E_m)$ in \mathcal{T} such that
 - 1 $\text{Hom}_{\mathcal{T}}(E_i, E_i) \simeq k$ for $i = 1, \dots, m$.

Strong exceptional sequences

Let k be a field and let $(\mathcal{T}, \Sigma, \Delta)$ be a k -linear triangulated category.

- ▶ A **strong exceptional sequence** is a sequence of objects $\mathcal{E} = (E_1, \dots, E_m)$ in \mathcal{T} such that
- 1 $\text{Hom}_{\mathcal{T}}(E_i, E_i) \simeq k$ for $i = 1, \dots, m$.
 - 2 $\text{Hom}_{\mathcal{T}}(E_i, E_j) = 0$ for all $i > j$ (no backward morphisms).

Strong exceptional sequences

Let k be a field and let $(\mathcal{T}, \Sigma, \Delta)$ be a k -linear triangulated category.

- ▶ A **strong exceptional sequence** is a sequence of objects $\mathcal{E} = (E_1, \dots, E_m)$ in \mathcal{T} such that
- 1 $\text{Hom}_{\mathcal{T}}(E_i, E_i) \simeq k$ for $i = 1, \dots, m$.
 - 2 $\text{Hom}_{\mathcal{T}}(E_i, E_j) = 0$ for all $i > j$ (no backward morphisms).
 - 3 $\text{Hom}_{\mathcal{T}}(E_i, \Sigma^\ell E_j) = 0$ for all i, j and all $\ell \neq 0$ (no higher morphisms).

Strong exceptional sequences

Let k be a field and let $(\mathcal{T}, \Sigma, \Delta)$ be a k -linear triangulated category.

- ▶ A **strong exceptional sequence** is a sequence of objects $\mathcal{E} = (E_1, \dots, E_m)$ in \mathcal{T} such that
 - ① $\text{Hom}_{\mathcal{T}}(E_i, E_i) \simeq k$ for $i = 1, \dots, m$.
 - ② $\text{Hom}_{\mathcal{T}}(E_i, E_j) = 0$ for all $i > j$ (no backward morphisms).
 - ③ $\text{Hom}_{\mathcal{T}}(E_i, \Sigma^\ell E_j) = 0$ for all i, j and all $\ell \neq 0$ (no higher morphisms).
- ▶ An exceptional sequence is called **full** if the smallest triangulated subcategory generated by \mathcal{E} , denoted by \mathcal{E}^Δ , equals \mathcal{T} .

Theorem: Equivalences induced by strong exceptional sequences

Let k be a field, \mathcal{P} a k -linear Hom-finite category and \mathcal{E} is strong exceptional sequence in $\mathcal{K}^b(\mathcal{P})$. Then, there exists an acyclic quiver $q_{\mathcal{E}}$ and set of relations $\rho \subset \text{Mor}(k\mathcal{F}_{q_{\mathcal{E}}})$ such that

$$\mathcal{D}^b(\text{rep}_k(q_{\mathcal{E}}, \rho)) \simeq \mathcal{K}^b(\mathcal{E}^{\oplus}) \simeq \mathcal{E}^{\Delta} \subseteq \mathcal{K}^b(\mathcal{P})$$

where \mathcal{E}^{\oplus} is the full subcategory of $\mathcal{K}^b(\mathcal{P})$ generated by all finite direct sums of objects in \mathcal{E} .

The convolution functor in one slide

The convolution functor is the functor $\mathcal{K}^b(\mathcal{E}^\oplus) \xrightarrow{F} \mathcal{E}^\Delta \subseteq \mathcal{K}^b(\mathcal{P})$. Let U be an object in $\mathcal{K}^b(\mathcal{E}^\oplus)$ with lower bound m and upper bound n . Then $F(U)$ can be computed as follows:

$$\dots \xrightarrow{\partial^{n-4}} U^{n-3} \xrightarrow{\partial^{n-3}} U^{n-2} \xrightarrow{\partial^{n-2}} U^{n-1} \xrightarrow{\partial^{n-1}} U^n \longrightarrow 0$$

The convolution functor in one slide

The convolution functor is the functor $\mathcal{K}^b(\mathcal{E}^\oplus) \xrightarrow{F} \mathcal{E}^\Delta \subseteq \mathcal{K}^b(\mathcal{P})$. Let U be an object in $\mathcal{K}^b(\mathcal{E}^\oplus)$ with lower bound m and upper bound n . Then $F(U)$ can be computed as follows:

- ▶ If $m = n$, then $F(U) := \Sigma^{-m}U^m \in \mathcal{E}^\Delta$.

$$\dots \xrightarrow{\partial^{n-4}} U^{n-3} \xrightarrow{\partial^{n-3}} U^{n-2} \xrightarrow{\partial^{n-2}} U^{n-1} \xrightarrow{\partial^{n-1}} U^n \longrightarrow 0$$

The convolution functor in one slide

The convolution functor is the functor $\mathcal{K}^b(\mathcal{E}^\oplus) \xrightarrow{F} \mathcal{E}^\Delta \subseteq \mathcal{K}^b(\mathcal{P})$. Let U be an object in $\mathcal{K}^b(\mathcal{E}^\oplus)$ with lower bound m and upper bound n . Then $F(U)$ can be computed as follows:

- ▶ If $m = n$, then $F(U) := \Sigma^{-m} U^m \in \mathcal{E}^\Delta$.

$$\begin{array}{ccccccc}
 \dots & \xrightarrow{\partial^{n-4}} & U^{n-3} & \xrightarrow{\partial^{n-3}} & U^{n-2} & \xrightarrow{\partial^{n-2}} & U^{n-1} & \xrightarrow{\partial^{n-1}} & U^n & \longrightarrow & 0 \\
 & & & & & & \uparrow \Delta & & \swarrow & & \\
 & & & & & & \Sigma^{-1}\pi(\partial^{n-1}) & & & & \\
 & & & & & & \downarrow & & \searrow & & \\
 & & & & & & \Sigma^{-1}\text{Cone}(\partial^{n-1}) & & & &
 \end{array}$$

The convolution functor in one slide

The convolution functor is the functor $\mathcal{K}^b(\mathcal{E}^\oplus) \xrightarrow{F} \mathcal{E}^\Delta \subseteq \mathcal{K}^b(\mathcal{P})$. Let U be an object in $\mathcal{K}^b(\mathcal{E}^\oplus)$ with lower bound m and upper bound n . Then $F(U)$ can be computed as follows:

- ▶ If $m = n$, then $F(U) := \Sigma^{-m} U^m \in \mathcal{E}^\Delta$.
- ▶ Compute \mathcal{X} with $\mathcal{X} \cdot \Sigma^{-1} \pi(\partial^{n-1}) = \partial^{n-2}$ and $\partial^{n-3} \cdot \mathcal{X} = 0$.
- ▶ Repeat the procedure for the following new object U :

$$\dots \xrightarrow{\partial^{n-4}} U^{n-3} \xrightarrow{\partial^{n-3}} U^{n-2} \xrightarrow{\mathcal{X}} \Sigma^{-1} \text{Cone}(\partial^{n-1}) \longrightarrow 0 \longrightarrow 0$$

The convolution functor in one slide

The convolution functor is the functor $\mathcal{K}^b(\mathcal{E}^\oplus) \xrightarrow{F} \mathcal{E}^\Delta \subseteq \mathcal{K}^b(\mathcal{P})$. Let U be an object in $\mathcal{K}^b(\mathcal{E}^\oplus)$ with lower bound m and upper bound n . Then $F(U)$ can be computed as follows:

- ▶ If $m = n$, then $F(U) := \Sigma^{-m} U^m \in \mathcal{E}^\Delta$.
- ▶ Compute \mathcal{X} with $\mathcal{X} \cdot \Sigma^{-1} \pi(\partial^{n-1}) = \partial^{n-2}$ and $\partial^{n-3} \cdot \mathcal{X} = 0$.
- ▶ Repeat the procedure for the following new object U :

$$\dots \xrightarrow{\partial^{n-4}} U^{n-3} \xrightarrow{\partial^{n-3}} U^{n-2} \xrightarrow{\mathcal{X}} \Sigma^{-1} \text{Cone}(\partial^{n-1}) \longrightarrow 0 \longrightarrow 0$$

Convolution functor also depends on solving two-sided linear systems.

References

- 1 S. Posur, *A constructive approach to Freyd categories*
- 2 S. I. Gelfand and Y. I. Manin, *Methods of Homological Algebra*
- 3 **CAP on GitHub:**
https://github.com/homalg-project/CAP_project
- 4 **HOMALG on GitHub:**
https://github.com/homalg-project/homalg_project
- 5 **HIGHERHOMOLOGICALALGEBRA on GitHub:** <https://github.com/homalg-project/HigherHomologicalAlgebra>

Thank you for your attention