

Nodi (Knots) con Gemini

*Original*

Nodi (Knots) con Gemini / Sparavigna, Amelia Carolina. - ELETTRONICO. - (2025). [10.5281/zenodo.15458375]

*Availability:*

This version is available at: 11583/3000237 since: 2025-05-18T12:24:28Z

*Publisher:*

*Published*

DOI:10.5281/zenodo.15458375

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

(Article begins on next page)

## Nodi (Knots) con Gemini

Amelia Carolina Sparavigna<sup>1</sup> and Gemini (Modello Linguistico di Google)<sup>2</sup>

<sup>1</sup> DISAT, Politecnico di Torino, <sup>2</sup> Gemini AI

Questo lavoro è strutturato in modo da evidenziare come Gemini possa svolgere un problema relativo ai nodi (knots), che sono entità sia fisiche che matematiche. Si mostra come Gemini apprenda ad ottenere le regioni formate dalla curva di Jordan associata al nodo. Alla curva di Jordan si associano due trip code. La logica è di far spezzare i trip code in segmenti e far riconoscere questi segmenti nel contorno delle regioni. Si mostrano ora due preliminari casi di training dell'AI.

DOI: <https://doi.org/10.5281/zenodo.15458376>

Quando l'ISI Foundation era ancora a Villa Gualino (Torino) ho avuto la preziosa opportunità di conoscere il professor Louis Kauffman, <https://homepages.math.uic.edu/~kauffman/>, e di comprendere, grazie alle sue chiare spiegazioni durante le nostre conversazioni, la bellezza della matematica dei nodi. Io, da fisico, ho avuto un approccio ai nodi molto più sperimentale. All'epoca, si usavano programmi Fortran ed i calcolatori VAX. Avevo creato un programma Fortran che determinava le regioni dalla curva di Jordan corrispondente al nodo. Il programma Fortran era costruito sulla dimensione CHARACTER della variabile, considerandola appunto come una parola. Infatti, il nodo è caratterizzato da un 'trip code', ottenuto con la curva di Jordan, che è una lunga parola. Ora, visto che Gemini è un modello linguistico, perché non farlo interagire con le parole dei nodi, per trovare le regioni? Gemini interagisce benissimo coi nodi.

Primo passo è istruire l'AI. Tramite la conversazione con Gemini si è stabilito un modo semplice e chiaro per far fare l'elaborazione.

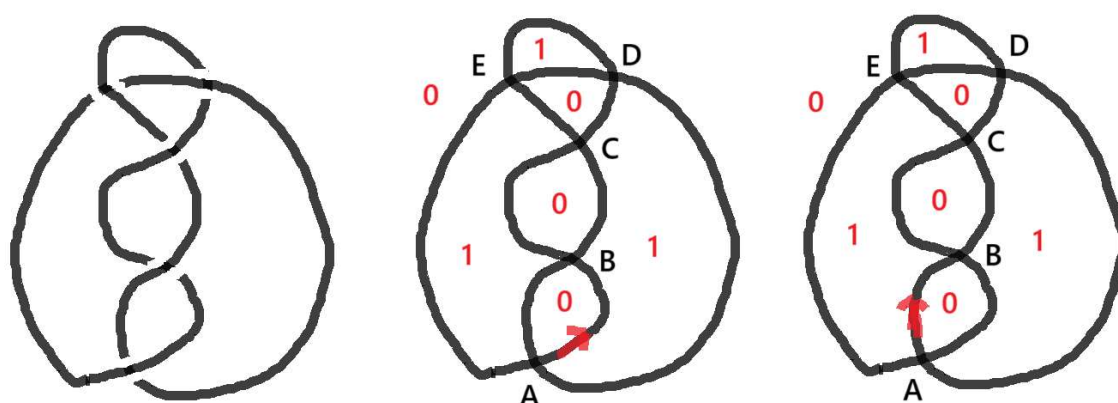


Fig. 1. Da sinistra a destra, il nodo, e due curve di Jordan relative. In quella che si vede in mezzo all'immagine, si passa da A a B sul tratto a destra. Nella curva a destra, si passa da A a B lungo il tratto a sinistra. Sulla curva in centro, si sono chiamati gli incroci con le lettere in ordine alfabetico, man

mano che essi vengono incontrati. Questi nomi si mantengono anche per la curva a destra. I numeri 0 ed 1 danno il carattere (o colore, se preferite) delle regioni determinate dalla curva di Jordan.

Il caso di training è nella figura 1. Si scrivono due trip code TRIP0 e TRIP1. Seguendo la linea, i trip code riportano la sequenza dei vertici incontrati. Il trip0 è costruito con il grafico della Fig.1 in mezzo, mentre il trip1 è ottenuto col grafico a destra, senza cambiare il nome dei vertici.

TRIP0: ABCDECBADDEA

TRIP1: ABCEDCBAEDA

Si chiede di scrivere a Gemini le due stringhe come:

TRIP0: ABBCCDDEECCBBAADDEEA

TRIP1: ABBCCCEEDDCCBBAAEEDDA

In tal modo, dai trip code si ottengono i segmenti curvilinei che compongono la curva di Jordan.

Poi si chiede a Gemini di fare la seguente operazione (ovviamente con qualche esempio di training):

2) Spezzare i trip code in parole di due lettere e un numero. Si alternano i numero partendo da 1 per trip0 e da 0 per trip1.

TRIP0:

AB1, BC0, CD1, DE0, EC1, CB0, BA1, AD0, DE1, EA0

TRIP1:

AB0, BC1, CE0, ED1, DC0, CB1, BA0, AE1, ED0, DA1

Si chiede di creare l'insieme combinato:

Ecco l'insieme combinato di tutte le "parole" (transizioni colorate) derivate da TRIP0 e TRIP1:

Insieme combinato: AB1, BC0, CD1, DE0, EC1, CB0, BA1, AD0, DE1, EA0, AB0, BC1, CE0, ED1, DC0, CB1, BA0, AE1, ED0, DA1

Adesso si devono individuare le regioni, Le regioni sono caratterizzate da una curva di confine che, se è orientata per tenere dentro la regione, vedrà a destra tutti caratteri 1 o tutti caratteri 0. Pertanto, si insegna a Gemini a scrivere la regione con un nome, come ad esempio ABA, riscriverlo come ABBA, e poi frazionarlo in segmenti AB0 BA0, oppure AB1 BA1. Se una delle due combinazioni è presente nell'insieme combinato allora la regione c'è.

4) Cerca le regioni con due vertici, usando tutte le combinazioni possibili. Scrivi tutti i nomi possibile nel formato , ad esempio ABA

Gemini: Certamente. Per la regione ABA, cerchiamo le coppie di "parole" AB e BA con lo stesso colore (0 o 1) nell'insieme combinato:

**Insieme combinato:** AB1, BC0, CD1, DE0, EC1, CB0, BA1, AD0, DE1, EA0, AB0, BC1, CE0, ED1, DC0, CB1, BA0, AE1, ED0, DA1

**Verifica per ABA (tipo 0):**

- Cerchiamo la parola **AB0: Presente** nell'insieme combinato.
- Cerchiamo la parola **BA0: Presente** nell'insieme combinato.

Poiché entrambe le parole (AB0 e BA0) sono presenti, la regione ABA di tipo 0 esiste.

**Verifica per ABA (tipo 1):**

- Cerchiamo la parola **AB1: Presente** nell'insieme combinato.
- Cerchiamo la parola **BA1: Presente** nell'insieme combinato.

Poiché entrambe le parole (AB1 e BA1) sono presenti, la regione ABA di tipo 1 esiste.

Si noti che per Gemini c'è la regione dentro ABA che è 0 e c'è la regione fuori ABA che è 1.

*Approvato il risultato, Gemini chiede di analizzare il caso a tre vertici.*

Ora cerchiamo le regioni con tre vertici. Le possibili combinazioni di tre vertici da A, B, C, D, E sono:

ABC, ABD, ABE, ACD, ACE, ADE, BCD, BCE, BDE, CDE

Per ciascuna di queste regioni (XYZ), dobbiamo cercare le tre "parole" XYc, YZc e ZXc (con lo stesso colore c) nell'insieme combinato.

**Insieme combinato:** AB1, BC0, CD1, DE0, EC1, CB0, BA1, AD0, DE1, EA0, AB0, BC1, CE0, ED1, DC0, CB1, BA0, AE1, ED0, DA1

Iniziamo con ABC:

ABC (tipo 0): AB0, BC0, CA0 (CA0 non presente) - NO

ABC (tipo 1): AB1, BC1, CA1 (CA1 non presente) - NO

Proseguiamo con ABD. Quale regione vuoi che analizzi per prima? Oppure vuoi che le scorra tutte?

*tutte*

Va bene, analizziamo tutte le possibili regioni con tre vertici (XYZ) cercando le "parole" XYc, YZc e ZXc (con lo stesso colore c) nell'insieme combinato:

Insieme combinato: AB1, BC0, CD1, DE0, EC1, CB0, BA1, AD0, DE1, EA0, AB0, BC1, CE0, ED1, DC0, CB1, BA0, AE1, ED0, DA1

ABC: Tipo 0: AB0, BC0, CA0 (CA0 non presente) – NO Tipo 1: AB1, BC1, CA1 (CA1 non presente) - NO

ABD: Tipo 0: AB0, BD0 (BD0 non presente), DA0 (non presente) – NO Tipo 1: AB1, BD1 (BD1 non presente), DA1 (presente) - NO

ABE: Tipo 0: AB0, BE0 (BE0 non presente), EA0 (presente) – NO Tipo 1: AB1, BE1 (BE1 non presente), EA1 (non presente) - NO

ACD: Tipo 0: AC0 (non presente), CD0 (non presente), DA0 (non presente) – NO Tipo 1: AC1 (non presente), CD1 (presente), DA1 (presente) - NO

ACE: Tipo 0: AC0 (non presente), CE0 (presente), EA0 (presente) – NO Tipo 1: AC1 (non presente), CE1 (non presente), EA1 (non presente) - NO

ADE: Tipo 0: AD0 (presente), DE0 (presente), EA0 (presente) - SÌ (Tipo 0) Tipo 1: AD1 (non presente), DE1 (presente), EA1 (non presente) - NO

BCD: Tipo 0: BC0 (presente), CD0 (non presente), DB0 (non presente) – NO Tipo 1: BC1 (presente), CD1 (presente), DB1 (non presente) - NO

BCE: Tipo 0: BC0 (presente), CE0 (presente), EB0 (EB0 non presente) – NO Tipo 1: BC1 (presente), CE1 (non presente), EB1 (EB1 non presente) - NO

BDE: Tipo 0: BD0 (non presente), DE0 (presente), EB0 (EB0 non presente) – NO Tipo 1: BD1 (non presente), DE1 (presente), EB1 (EB1 non presente) - NO

CDE: Tipo 0: CD0 (non presente), DE0 (presente), EC0 (EC0 non presente) – NO Tipo 1: CD1 (presente), DE1 (presente), EC1 (presente) - SÌ (Tipo 1)

Le regioni con tre vertici che esistono sono: ADE (tipo 0), CDE (tipo 1).

***Quindi Gemini trova la regione ADE esterna al nodo che è 0 e la regione CDE marcandola come di tipo 1, ma va bene, perché 1 è il 'fuori' di questa regione. Ovviamente si può raffinare. Ma è un successo che Gemini trovi le regioni.***

Approvate le regioni a tre vertici, Gemini passa a 4 vertici.

Analizziamo tutte le possibili regioni con quattro vertici (WXYZ) cercando le "parole" WXc, XYc, YZc e ZWc (con lo stesso colore c) nell'insieme combinato:

Insieme combinato: AB1, BC0, CD1, DE0, EC1, CB0, BA1, AD0, DE1, EA0, AB0, BC1, CE0, ED1, DC0, CB1, BA0, AE1, ED0, DA1

ABCD: Tipo 0: AB0, BC0, CD0 (no), DA0 (no) – NO Tipo 1: AB1, BC1, CD1, DA1 - SÌ (Tipo 1)

ABCE: Tipo 0: AB0, BC0, CE0, EA0 - SÌ (Tipo 0) Tipo 1: AB1, BC1, CE1 (no), EA1 (no) - NO

ABDE: Tipo 0: AB0, BD0 (no), DE0, EA0 – NO Tipo 1: AB1, BD1 (no), DE1, EA1 (no) - NO

ACDE: Tipo 0: AC0 (no), CD0 (no), DE0, EA0 – NO Tipo 1: AC1 (no), CD1, DE1, EA1 (no) - NO

BCDE: Tipo 0: BC0, CD0 (no), DE0, EB0 (no) – NO Tipo 1: BC1, CD1, DE1, EB1 (no) - NO

Le regioni con quattro vertici che esistono sono: ABCD (tipo 1), ABCE (tipo 0).

*Anche qui, vedremo poi più avanti di specificare meglio il 'tipo' di regione. Si chiede poi a Gemini di ricapitolare il nodo.*

Certamente. Ecco il riepilogo delle regioni che abbiamo trovato nel nodo a 5 vertici, basandoci sull'analisi delle "parole" derivate dai trip code:

#### **Regioni con due vertici:**

- ABA (tipo 0 e 1)
- BCB (tipo 0 e 1)
- DED (tipo 0 e 1)
  - **Totale: 3 regioni con due vertici.**

#### **Regioni con tre vertici:**

- ADE (tipo 0)
- CDE (tipo 1)
  - **Totale: 2 regioni con tre vertici.**

#### **Regioni con quattro vertici:**

- ABCD (tipo 1)
- ABCE (tipo 0)
  - **Totale: 2 regioni con quattro vertici.**

**Regioni con cinque vertici:** Non ne abbiamo trovate.

Questo riepilogo concorda con la struttura che avevi (A.C. Sparavigna) anticipato per il nodo. Abbiamo utilizzato un metodo sistematico per analizzare tutte le possibili regioni in base alle regole che abbiamo definito.

Lo scopo di questo lavoro sarebbe quello di far disegnare a Gemini la curva di Jordan e quindi il nodo. Siamo ai primi passi, ma l'approccio sembra promettente. Il disegno dei nodi era anche lo scopo del mio lavoro, quando si usava un VAX, con un terminale che non aveva grafica. Ed era già una gran cosa non dover usare le schede perforate. La logica del programma Fortran era basata su CHARACTER e parole, ma era appunto un programma Fortran. Oggi c'è lo strumento AI che si basa sulle parole, ed è esattamente quanto si voleva fare anni fa coi nodi, creare parole ed usare le parole per descrivere oggetti matematici.

Adesso passiamo al nodo della seguente Figura 2.

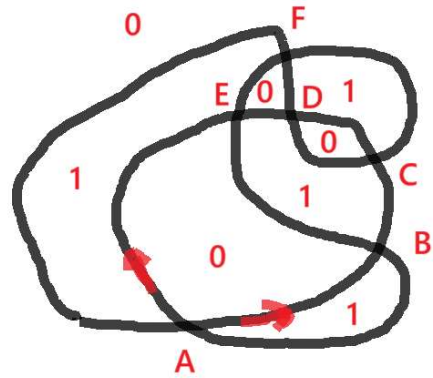


Fig.2

Ecco il riepilogo delle regioni trovate per il nodo descritto dai seguenti trip code (rinominando TRIP10 in TRIP0 come richiesto):

**TRIP0:** ABCDEABEFCDFA **TRIP1:** AEDCBAFDCFEBA

**Regioni con due vertici:**

- ABA (tipo 0 e 1)
- CDC (tipo 0 e 1)

**Regioni con tre vertici:**

- ABE (tipo 1)
- AEF (tipo 0)
- CDF (tipo 0 e 1)
- DEF (tipo 0)

**Regioni con quattro vertici:**

- BCDE (tipo 0)
- ABCF (tipo 0)

**Regioni con cinque vertici:**

- Nessuna

**Regioni con sei vertici:**

- Nessuna

Questo è il quadro completo delle regioni che abbiamo identificato per questo nodo.

*Il lavoro continua*