General “Definition”: Structural Index

- “Importance” has many aspects but minimal def. for centrality: Only depends on structure of graph.

- Structural Index: Let $G = (V,E,w)$ be a weighted directed or undirected multigraph. A function $s: V \rightarrow \mathbb{R}$ (or $s: E \rightarrow \mathbb{R}$) is a structural index iff

$$\forall x: G \cong H \rightarrow s_G(x) = s_H(\phi(x))$$

(Recall: Two graphs $G$ and $H$ are isomorphic ($G \cong H$) iff exists a bijective mapping $\phi: G \rightarrow H$ so that $(u,v) \in E$ iff $(\phi(u),\phi(v)) \in H$)

- structural index induces (total) partial-order (≤) on nodes/edges

- centrality can usually only be viewed as measured on an ordinal scale only (not interval or ratio scale)
**General “Definition”: Structural Index**

- Importance has many aspects but minimal def. for centrality: Only depends on structure of graph.

- **Structural Index**: Let $G = (V, E, w)$ be a weighted directed or undirected multigraph. A function $s: V \rightarrow \mathbb{R}$ (or $s: E \rightarrow \mathbb{R}$) is a structural index iff

$$\forall x: G \cong H \rightarrow s_G(x) = s_H(\phi(x))$$

(Recall: Two graphs $G$ and $H$ are isomorphic ($G \cong H$) iff exists a bijective mapping $\Phi: G \rightarrow H$ so that $(u, v) \in G$ iff $(\Phi(u), \Phi(v)) \in H$)

- structural index induces (total) partial-order ($\preceq$) on nodes/edges

- $\rightarrow$ centrality can usually only be viewed as measured on an ordinal scale only (not interval or ratio scale)

**Distance- and Neighborhood-based Centralities**

- Centrality-measures defined on the basis of distances or neighbourhoods in the graph:

  Centrality of vertex $\leftrightarrow$ “reachability” of a vertex

- **Neighborhoods: Degree Centrality**

  - Most basic: Degree centrality: $c(u) = \deg(u)$ (or $c(u) = \text{in-deg}(u)$ or $c(u) = \text{out-deg}(u)$) $\rightarrow$ local measure

  - Applicable: If edges have “direct vote” semantics

**Distances: Eccentricity**

- **Eccentricity** $e(u) = \max\{d(u, v); v \in V\}$

- Center of a graph: Set of all nodes with minimum eccentricity

- Eccentricity based centrality measure:

$$c(u) = \frac{1}{e(u)} = \frac{1}{\max\{d(u, v); v \in V\}}$$

- $\rightarrow$ nodes in the center of the graph have maximal centrality

**Distances: Eccentricity**

- **Eccentricity** $e(u) = \max\{d(u, v); v \in V\}$

- Center of a graph: Set of all nodes with minimum eccentricity

- Eccentricity based centrality measure:

$$c(u) = \frac{1}{e(u)} = \frac{1}{\max\{d(u, v); v \in V\}}$$

- $\rightarrow$ nodes in the center of the graph have maximal centrality
Distances: Eccentricity

- **Eccentricity** \( e(u) = \max\{d(u,v); v \in V\} \)
- **Center** of a graph: Set of all nodes with minimum eccentricity

Eccentricity based centrality measure:
\[
c(u) = \frac{1}{e(u)} = \frac{1}{\max \{d(u,v); v \in V\}}
\]

- \( \rightarrow \) nodes in the center of the graph have maximal centrality 😊

Distances: Closeness

- **Eccentricity** \( e(u) = \max\{d(u,v); v \in V\} \)
- **Center** of a graph: Set of all nodes with minimum eccentricity

Eccentricity based centrality measure:
\[
c(u) = \frac{1}{e(u)} = \frac{1}{\max \{d(u,v); v \in V\}}
\]

- \( \rightarrow \) nodes in the center of the graph have maximal centrality 😊

Minumum problem: find nodes whose sum of distances to other nodes is minimal (⇒ service facility location problem): For all \( u \) minimize total sum of minimal distances \( \sum_{v \in V} d(u,v) \)

Social analog: Determine central figure for coordination tasks

Example:

- Example graph with \( \sum_{v \in V} d(u,v) \) values

- Example graph with \( e(u) \) values
Distances: Closeness

- Minimun problem: find nodes whose sum of distances to other nodes is minimal (→ service facility location problem). For all u minimize total sum of minimal distances \( \sum_{v \in \mathcal{V}} d(u,v) \)

- Social analog: Determine central figure for coordination tasks

- Example:

![Graph with \( \sum_{v \in \mathcal{V}} d(u,v) \) values]

```
36  26  24  22  32
```

Distances: Closeness

- Possible resulting centrality index: closeness:

\[
c(u) = \sum_{v \in \mathcal{V}} \frac{1}{d(u,v)}
\]

Only applicable to connected graphs; disconnected graph: all nodes will get the same centrality \( 1/\infty \)

- Other possibility

\[
c(u) = \sum_{v \in \mathcal{V}} \left( \Delta_G + 1 - d(u,v) \right) \frac{1}{|\mathcal{V}| - 1}
\]

\( \Delta_G \) is the diameter of the graph

- If computed on directed graph: (set \( d(u,u) = 0 \) and set \( d(u,v) = 0 \) if \( u,v \) are unreachable via directed path → problematic!); using in-distances: „integration“, using out-distances „radiality“ (see [6])
Distances: Closeness

- Possible resulting centrality index: closeness:

\[ c(u) = \frac{1}{\sum_{v \in \mathcal{V}} d(u,v)} \]

Only applicable to connected graphs; disconnected graph: all nodes will get the same centrality \(1/\infty\)

- Other possibility

\[ c(u) = \frac{\sum_{v \in \mathcal{V}} (\Delta_G + 1 - d(u,v))}{|V| - 1} \]

\(\Delta_G\) is the diameter of the graph

- If computed on directed graph: (set \(d(u,u) = 0\) and set \(d(u,v) = 0\) if \(u, v\) are unreachable via directed path \(\rightarrow\) problematic \(\leftarrow\)); using in-distances: "integration", using out-distances "radiality" (see [6])

---

Distances: Closeness

- Possible resulting centrality index: closeness:

\[ c(u) = \frac{1}{\sum_{v \in \mathcal{V}} d(u,v)} \]

Only applicable to connected graphs; disconnected graph: all nodes will get the same centrality \(1/\infty\)

- Other possibility

\[ c(u) = \frac{\sum_{v \in \mathcal{V}} (\Delta_G + 1 - d(u,v))}{|V| - 1} \]

\(\Delta_G\) is the diameter of the graph

- If computed on directed graph: (set \(d(u,u) = 0\) and set \(d(u,v) = 0\) if \(u, v\) are unreachable via directed path \(\rightarrow\) problematic \(\leftarrow\)); using in-distances: "integration", using out-distances "radiality" (see [6])
**Competitive objective**: Given number of competitors: where to open a store (Customers will just choose store based on minimal distance)?

**Social Problem**: Example: find “social ecological niche”

**Formalization**: For \( u, v : \gamma_u(v) = \text{number of vertices closer to } u \text{ than to } v \); if one salesman selects \( u \) and competitor selects \( v \) as locations, the first will have

\[
\gamma_u(v) + \frac{1}{2} (|V| - \gamma_u(v) - \gamma_u(u)) = \frac{1}{2} |V| + \frac{1}{2} (\gamma_u(v) - \gamma_u(u))
\]

customers
**Competitive objective:** Given number of competitors: where to open a store (Customers will just choose store based on minimal distance)?

**Social Problem:** Example: find “social ecological niche”

**Formalization:** For $u, v : \gamma_u(v) =$ number of vertices closer to $u$ than to $v$; if one salesman selects $u$ and competitor selects $v$ as locations, the first will have

$$\gamma_u(v) + \frac{1}{2} (|V| - \gamma_u(v) - \gamma_v(u)) = \frac{1}{2} |V| + \frac{1}{2} (\gamma_u(v) - \gamma_v(u))$$

Customers

→ Competitor will want to minimize

$$f(u, v) = \gamma_u(v) - \gamma_v(u)$$

→ Possible centrality index: First salesman knows the strategy of the competitor and calculates for each location the worst case:

$$c(u) = \min_v \{ f(u, v) : v \in V / \{u\} \}$$

$c(u)$ is called centroid value: measures the advantage of location $u$ compared to other locations: Minimal loss of customers if he choses $u$ and a competitor chooses $v$
Distances: Centroids

- Competitor will want to minimize
  \[ f(u, v) = y_u(v) - y_u(u) \]

- **Possible centrality index:** First salesman knows the strategy of the competitor and calculates for each location the worst case:
  \[ c(u) = \min_v \{ f(u, v) : v \in V \setminus \{u\} \} \]

- \( c(u) \) is called centroid value: measures the advantage of location \( u \) compared to other locations: Minimal loss of customers if he chooses \( u \) and a competitor chooses \( v \).

Shortest Paths: Shortest Path Betweenness

- Again assume that communication (workflows etc.) happen along shortest paths only. Let
  \[ \delta_{ab}(v) = \frac{\sigma_{ab}(v)}{\sigma_{ab}} \]

  with \( \sigma_{ab} \): total number of shortest paths between nodes \( a \) and \( b \).

  **Interpretation:** Probability that \( v \) is involved in a communication between \( a \) and \( b \).

- **Shortest Path Betweenness (SPB) centrality** is then:
  \[ c(v) = \sum_{a \in V} \sum_{b \in V} \delta_{ab}(v) \]

- **Interpretation:** Control that \( v \) exerts on the communication in the graph.

- Also applicable to disconnected graphs.

- Algorithm by Ulrik Brandes computes SPB in \( O(|V||E| + |V|^2 \log |V|) \) time.
Shortest Paths: Shortest Path Betweenness

- Define $c_{\text{SPB}}$ for edges analogously:
  \[ c(e) = \sum_{v \in V} \sum_{b \in V} \delta_{ab}(e) \]

- **Possible:** Interpret quantity $\delta_{ab}(v)$ as general relative information flow through $v$ ("rush")

- **Other variants:** Instead of shortest paths between $a$ and $b$ regard:
  - the set of all paths
  - the set of the $k$-shortest paths (interesting for social case; choose small $k$)
  - the set of the $k$-shortest node disjoint paths
  - the set of paths not longer than $(1+\varepsilon)d(a,b)$

Deriving edge centralities from vertex centralities

- **What we have seen so far:** Various centrality measures mostly for vertices (based on degree, closeness, betweenness)

- **→ Formal way to translate a given vertex centrality index to a corresponding edge centrality:** Apply the given vertex centrality to a transformed version of $G$, the edge graph

- Given original $G = (V,E)$ then the **edge graph** $G' = (E,K)$ is defined by taking original edges as vertices. Two original edges are connected in $G'$ if they are originally incident to the same original node.

- **Size of $G'$** may be quadratic (w.r.t. number of nodes) compared to $G$

Deriving edge centralities from vertex centralities

- **Remember:** Vertex stress centrality for node $x$: Number of shortest paths that use $x$; Straightforward version for edge $e$: Number of shortest paths that use $e$;

- **→ Upper Example:** $G$: Stress centrality of edge $a$ would be 3; But in edge graph $G'$ stress centrality of original edge $a$ (now a node) is 0.

- **→ Formal translations of vertex centrality indices to edge centralities with edge graphs are not well suited for all purposes**

- **→ Introduce incidence graph $G''$:** Each original edge is split by new “edge vertex” that represents the edge → Now vertex indices can be applied, preserving the intuition.
Deriving edge centralities from vertex centralities

- **Remember**: Vertex stress centrality for node \( x \): Number of shortest paths that use \( x \); Straightforward version for edge \( e \): Number of shortest paths that use \( e \);

- **Upper Example**: \( G \): Stress centrality of edge \( a \) would be 3; But in edge graph \( G' \) stress centrality of original edge \( a \) (now a node) is 0.

- **Formal translations of vertex centrality indices to edge centralities with edge graphs are not well suited for all purposes**

- **Introduce incidence graph \( G'' \)**: Each original edge is split by new "edge vertex" that represents the edge \( \rightarrow \) Now vertex indices can be applied, preserving the intuition.

Vitality

- **Intuition**: Measure importance of vertex (or edge) by the difference of a given quality measure \( q \) on \( G \) with or without the vertex (edge):

  - **Vitality** \( v(x) \) of graph element \( x \): \( v(x) = q(G) - q(G \setminus \{x\}) \)

- **Example 1 for quality measure \( q \): Flow**:

  - Given directed graph \( G \) with positive edge weights \( w \) modeling capacities. The flow \( f(s,t) \) from node \( s \) (source) to node \( t \) (sink) is defined as:

    \[
    f(s,t) = \sum_{e \in \text{Out-Edges of } s} \tilde{f}(e) - \sum_{e \in \text{In-Edges of } t} \tilde{f}(e)
    \]

  where the local flows \( \tilde{f} \) respect capacity constraints: \( 0 \leq \tilde{f}(e) \leq w(e) \)

  and balance conditions:

  \[
  \forall v \in V \setminus \{s,t\}: \sum_{e \in \text{Out-Edges of } v} \tilde{f}(e) = \sum_{e \in \text{In-Edges of } v} \tilde{f}(e)
  \]
- Intuition: Measure importance of vertex (or edge) by the difference of a given quality measure $q$ on $G$ with or without the vertex (edge):
  - $\rightarrow$ Vitality $v(x)$ of graph element $x: v(x) = q(G) - q(G\setminus x)$

- Example 1 for quality measure $q$: Flow:
  - Given directed graph $G$ with positive edge weights $w$ modeling capacities. The flow $f(s,t)$ from node $s$ (source) to node $t$ (sink) is defined as:

$$f(s,t) = \sum_{e \in \text{Out-Edges of } s} \tilde{f}(e) - \sum_{e \in \text{In-Edges of } t} \tilde{f}(e)$$

where the local flows $\tilde{f}$ respect capacity constraints: $0 \leq \tilde{f}(e) \leq w(e)$ and balance conditions:

$$\forall v \in V \setminus \{s,t\}: \sum_{e \in \text{Out-Edges of } v} \tilde{f}(e) = \sum_{e \in \text{In-Edges of } v} \tilde{f}(e)$$

- Computing a flow $f: E \rightarrow \mathbb{R}$ of maximum value (tweaking the local flows): $O(\log(|V|^2|E|))$ (Algorithm by Goldberg & Tarjan (see [2]))

- Now define quality measure by e.g.:

$$q(G) = \sum_{s,t \in V} \max f(s,t)$$

Vitality

- Computing a flow \( f : E \rightarrow \mathbb{R} \) of maximum value (tweaking the local flows): \( O(|V| |E| \log(|V|^2 |E|)) \) (Algorithm by Goldberg & Tarjan (see [2]))

- Now define quality measure by e.g.:

\[
q(G) = \sum_{s, t \in V} \max_I f(s, t)
\]


Vitality

- Intuition: Measure importance of vertex (or edge) by the difference of a given quality measure \( q \) on \( G \) with or without the vertex (edge):

\[
\rightarrow \text{Vitality } v(x) \text{ of graph element } x : v(x) = q(G) - q(G\setminus\{x\})
\]

- Example 1 for quality measure \( q \): Flow:

  - Given directed graph \( G \) with positive edge weights \( w \) modeling capacities. The flow \( f(s, t) \) from node \( s \) (source) to node \( t \) (sink) is defined as:

\[
f(s, t) = \sum_{e \in \text{Out-Edges of } s} \tilde{f}(e) = \sum_{e \in \text{In-Edges of } t} \tilde{f}(e)
\]

  where the local flows \( \tilde{f} \) respect capacity constraints: \( 0 \leq \tilde{f}(e) \leq w(e) \)

  and balance conditions:

\[
\forall v \in V \setminus \{s, t\} : \sum_{e \in \text{Out-Edges of } v} \tilde{f}(e) = \sum_{e \in \text{In-Edges of } v} \tilde{f}(e)
\]

Stress Centrality as Vitality

- Possible Interpretation: Distance \( d(v, w) \) represents costs to send message from \( v \) to \( w \)

- If \( x \) is a cut-vertex or bridge-edge \( \rightarrow \) Graph is disconnected after removal \( \rightarrow \) centrality cannot be computed.

- We had: stress centrality of \( v \) or \( e \) is equal to number of shortest paths through \( v \) or \( e \)

\[
c_{\text{stress}}(v) = \sum_{u \in \text{shortest-paths } v} \sum_{b \in \text{shortest-paths } v} \sigma_{ab}(v) \\
c_{\text{stress}}(e) = \sum_{a \in V \setminus v} \sum_{b \in V \setminus v} \sigma_{ab}(e)
\]

- Intuition: \( c_{\text{stress}}(v) \) seems to measure the number of shortest paths that would be lost if \( v \) wasn’t available any more

- Why can’t we directly use \( c_{\text{stress}} \) as a graph quality index to construct a vitality index?

\[
\rightarrow \text{Because actual number of shortest paths can INCREASE if e.g. edge is taken away}
\]
Stress Centrality as Vitality

- In order to define a vitality-like version of stress: Consider only those shortest paths that haven’t changed their length:

\[ c_{\text{vitality}}(v, G) = c_{\text{stress}}(v, G) - c_{\text{stress}}(v, G \setminus \{v\}) \]

with

\[ c_{\text{stress}}(v, G \setminus \{v\}) = \sum_{a \in \mathcal{A}, a \neq v} \sum_{b \in \mathcal{B}, b \neq v} \sigma_{ab}[d_G(a, b) = d_{G \setminus \{v\}}(a, b)] \]

(Iverson notation)

Critique on Betweenness Based Centralities

- major critique: Max-Flow betweenness centrality (suggested to counteract this drawback) may exhibit similar problems

- here: special Max-Flow betweenness centrality mfb:
  -- limit edge capacity to one
  -- mfb\((i)\) := maximum possible flow through \(i\) over all possible solutions to the s-t-maximum flow problem, averaged over all s and t.

(b) In calculations of flow betweenness, vertices A and B in this configuration will get high scores while vertex C will not.

Source: [5]
Random Walk Centrality == Current Flow Btw. Centrality (see [5])

- flow of electric current in a resistor network; Vᵢ = voltage (potential) at vertex i

- Current Flow betweenness cfb centrality: cfb(i) := amount of current that flows through i in this setup, averaged over all s and t.

---

Random Walk Centrality == Current Flow Btw. Centrality (see [5])

- flow of electric current in a resistor network; Vᵢ = voltage (potential) at vertex i

- Current Flow betweenness cfb centrality: cfb(i) := amount of current that flows through i in this setup, averaged over all s and t.
Random Walk Centrality == Current Flow Btw. Centrality (see [5])

Kirchhoff’s point law (current conservation): total current flow in / out of node is zero:

\[ \sum_j A_{ij}(V_i - V_j) = \delta_{is} - \delta_{it}, \]

if there is an edge between \( i \) and \( j \), otherwise,
\[ A_{ij} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases} \]

\[ \delta_j = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases} \]

one unit of current in

\[ s \]

one unit of current out

\[ \sum_j A_{ij} = k_i, \text{ the degree of vertex } i \]

\[ \sum_j A_{ij}(V_i - V_j) = \delta_{is} - \delta_{it} \quad \xrightarrow{\text{“Graph Laplacian”}} \quad (D - A) \cdot V = s \]

\( D \) is the diagonal matrix with elements \( D_{ii} = k_i \)

source vector \( s \) \[ s_i = \begin{cases} +1 & \text{for } i = s, \\ -1 & \text{for } i = t, \\ 0 & \text{otherwise.} \end{cases} \]

\[ V = (D - A)^{-1} \cdot s \]
\[ \sum_j A_{ij} = k_i, \text{ the degree of vertex } i \]

\[ \sum_j A_{ij}(V_i - V_j) = \delta_{is} - \delta_{it} \quad \longrightarrow \quad \begin{bmatrix} D - A \end{bmatrix} \cdot \begin{bmatrix} V \end{bmatrix} = \begin{bmatrix} s \end{bmatrix} \]

"Graph Laplacian"

D is the diagonal matrix with elements \( D_{ii} = k_i \)

source vector \( s \)
\[ s_i = \begin{cases} +1 & \text{for } i = s, \\ -1 & \text{for } i = t, \\ 0 & \text{otherwise}. \end{cases} \]

\[ V = (D_v - A_v)^{-1} \cdot s \]

\[ (D - A) \cdot V = s \]

Laplacian is not invertible, \( \det = 0 \), because system of eq. is overdetermined \( \Rightarrow \) set one \( V_v = 0 \) and measure voltages relative to \( v \). \( \Rightarrow \) remove the \( v \)-th row and column (since \( V_v = 0 \)) \( \Rightarrow \) now invertible

\[ V = (D_v - A_v)^{-1} \cdot s \quad \text{(matrix inversion: } O(n^3)) \]

let us now add a \( v \)-th row and column back into \( (D_v - A_v)^{-1} \) with values all equal to zero.

The resulting matrix we will denote \( T \).

\[ V^{(st)} = T_{is} - T_{it} \]

\[ \text{current flow at node } i: \quad I_i^{(st)} = \frac{1}{2} \sum_j A_{ij} |V_i^{(st)} - V_j^{(st)}| \]
Random Walk Centrality == Current Flow Btw. Centrality (see [5])

\[(D - A) \cdot V = s\]

Laplacian is not invertible, det = 0, because system of eq. is over determined \( \Rightarrow \) set one \( V_\nu = 0 \) and measure voltages relative to \( v \). \( \Rightarrow \) remove the \( v \)-th row and column (since \( V_v = 0 \)) \( \Rightarrow \) now invertible

\[V = (D_v - A_v)^{-1} \cdot s\]  (matrix inversion: \( O(n^3) \))

Let us now add a \( \nu \)-th row and column back into \((D_v - A_v)^{-1}\) with values all equal to zero.
The resulting matrix we will denote \( T \).

\[ V_i^{(st)} = T_{is} - T_{it} \]

\[ \text{current flow at node } i: \quad I_i^{(st)} = \frac{1}{2} \sum_j A_{ij} |V_i^{(st)} - V_j^{(st)}| \]

\[ \text{current flow at node } i: \quad I_i^{(st)} = \frac{1}{2} \sum_j A_{ij} |V_i^{(st)} - V_j^{(st)}| \]

\[ \text{unit current flow at nodes } s \text{ and } t: \quad I_s^{(st)} = 1, \quad I_t^{(st)} = 1. \]

\[ \text{cfsb}(i) \text{ (denoted as } b_i \text{) is then:} \]

\[ b_i = \frac{\sum_{s < t} I_i^{(st)}}{\frac{1}{2} n(n - 1)} \]

Example ([5])

Network 1

Network 2

<table>
<thead>
<tr>
<th>network</th>
<th>betweenness measure</th>
<th>shortest-path</th>
<th>max-flow</th>
<th>random walk</th>
<th>current flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network 1: vertices A &amp; B</td>
<td>0.636</td>
<td>0.631</td>
<td>0.670</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vertex C</td>
<td>0.200</td>
<td>0.282</td>
<td>0.333</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vertices X &amp; Y</td>
<td>0.200</td>
<td>0.068</td>
<td>0.269</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network 2: vertices A &amp; B</td>
<td>0.265</td>
<td>0.269</td>
<td>0.321</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vertex C</td>
<td>0.243</td>
<td>0.004</td>
<td>0.267</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vertices X &amp; Y</td>
<td>0.125</td>
<td>0.024</td>
<td>0.194</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Example ([5])

<table>
<thead>
<tr>
<th>Network</th>
<th>Betweenness Measure</th>
<th>Shortest-Path</th>
<th>Max-Flow</th>
<th>Random Walk / Current-Flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network 1: vertices A &amp; B</td>
<td>0.636</td>
<td>0.631</td>
<td>0.670</td>
<td></td>
</tr>
<tr>
<td>vertex C</td>
<td>0.200</td>
<td>0.282</td>
<td>0.333</td>
<td></td>
</tr>
<tr>
<td>vertices X &amp; Y</td>
<td>0.200</td>
<td>0.068</td>
<td>0.269</td>
<td></td>
</tr>
<tr>
<td>Network 2: vertices A &amp; B</td>
<td>0.265</td>
<td>0.269</td>
<td>0.321</td>
<td></td>
</tr>
<tr>
<td>vertex C</td>
<td>0.243</td>
<td>[0.004]</td>
<td>0.267</td>
<td></td>
</tr>
<tr>
<td>vertices X &amp; Y</td>
<td>0.125</td>
<td>0.024</td>
<td>0.194</td>
<td></td>
</tr>
</tbody>
</table>