# Paths with Jumps: Definition, Topology-preserving Dynamics, and Applications

**JADERICK P. PABICO**
*jppabico@uplb.edu.ph*
Institute of Computer Science, University of the Philippines Los Baños
PHILIPPINES

***Abstract*** *– Goggle and other list-based web services like Yahoo, Twitter and Facebook present their n-item results in m pages, where* $m \geq 1$. *The list can be represented as a path* $P_n(V, E)$ *of order n and size* $n – 1$, *where* $P_n$ *is particularly a sequence of n distinct nodes* $v_0, v_1, ..., v_{n-1} \in V$ *and* $(n – 1)$ *links* $(0, 1), (1, 2), ..., (n–2, n–1) \in E$. *When users are not satisfied with the first few nodes, they will conduct a walk W, often with jumps, that goes back and forth along* $P_n$ *until they find the kth node,* $0 \leq k < n$, *that satisfies their search. The simple walk can be modeled as a sequence of non-distinct nodes in* $P_n$, *but modeling it with jumps is difficult because there are no links in* $P_n$ *to allow the jumps between two non-adjacent nodes. To model this user behavior, the path is alternatively modeled as a sequence of m* $(m/n–1)$*-power of* $P_{m/n}$*'s separated by* $P_2$*'s, which is termed here as a path with jumps* $J_{m,n}$. *With this new representation for the list, W can be modeled as a simple walk over* $J_{m,n}$.

*In reality, the final walk W is an evolution of the time-progressed s sub-walks* $W_{(0)}, W_{(1)}, ..., W_{(s-1)}$, *where* $W_{(t)}$ *is a walk that was developed up to time* $t < s$. *This means that* $W_{(q)} \subset W_{(r)}$ *or* $W_{(q)} \sqsubset W_{(r)}$, $\forall q < r$, *where the symbols* $\subset$ *and* $\sqsubset$ *represent the subset and prefix relations, respectively. These relations are true when* $J_{m,n}$ *is static at best or under steady state at worst. However,* $J_{m,n}$ *is likewise dynamic. In this paper, a path with jumps as a special graph is introduced, the analysis of its dynamics is presented, and its possible application to modeling user behavior in most popular web services is discussed.*

**Keywords** – paths, walks, jumps, user-behavior modeling, web services

## I. INTRODUCTION

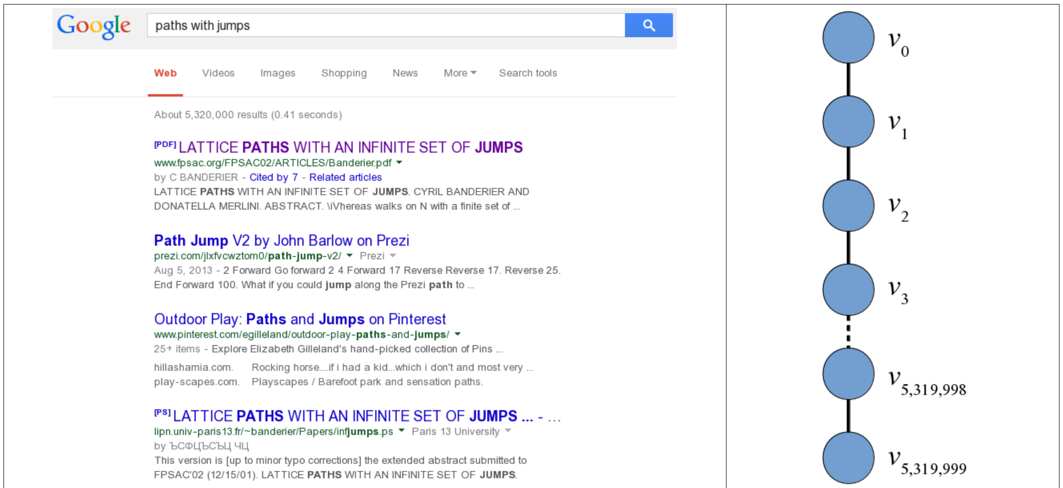
The start of the new millennium has seen the proliferation of various online services as a result of the ever-improving information and communication technologies (ICT). In fact, one among the many products of the so called ICT-boom is the Internet, which became in recent years a seemingly ubiquitous yet pervasive in the lives of humans who use it. Humans, in their natural tendency to be with fellow humans, constantly strive to be connected with each other by defying their physical spatial differences through ICT. Examples of online services that have gained global popularity, among many others, are Google® (www.google.com), Yahoo!® (www.yahoo.com), Facebook® (www.facebook.com), and Twitter® (www.twitter.com). These services provide the Internet users a list of information that, through a computer program, is either "pulled" by the users from the providers or "pushed" automatically by the providers to the users. Pulled information means that the users initiate the request for information from the providers, while pushed information means that the providers send the information without the explicit initiation from the users.

In cases of Google and Yahoo!, the user-pulled output is a list of items that the providers thought as relevant suggestions to the users' requests for information. In cases of Facebook and Twitter, the provider-pushed output is a list of posts of other users presented in a non-increasing chronological order. In any of these services, the list can be intuitively seen as a "graph" (see Definition 1), while the adjacencies between any two items in the list, for all items, is represented as a 0-1 matrix called the adjacency matrix (see Definition 2).

**Definition 1**. A <u>graph</u> $G(V,E)$ of order $n$ and size $m$ is composed of a set of $n$ nodes $V = \{v_0, v_1, \ldots, v_{n-1}\}$ and a set of $m$ links $E = \{(j, k)| vj, vk \in V\}$ that define the adjacency of any two nodes in $V$ [1].

**Definition 2**. The <u>adjacency matrix</u> of a graph $G$ of order $n$ is a 0-1 matrix $A$ whose $(j, k)$th element $A_{j,k}$ is 0 if nodes $v_j$ and $v_k$ are not adjacent to each other (i.e., there does not exist a link that connects them), and $A_{j,k}$ is 1 if nodes $v_j$ and $v_k$ are adjacent. For an undirected $G$, $m = ½ \sum_j \sum_k A_{j,k}$, while for a directed $G$, $m = \sum_j \sum_k A_{j,k}$ [1].

In this paper, the nodes were treated as the items in the list, while the links represent the adjacency of the nodes. The $n$-item result is usually presented as a non-increasing order of relevance to the pulled or pushed information. Thus, in the graph representation of the list, $v_0$ is the first item touted to be the one with the highest relevance from among the $n$ items. This is followed by $v_1$ with the second highest relevance, and so on up to $v_{n-1}$ which represents the item with the lowest relevance. The adjacency between the $j$th and the $(j+1)$th items are connected by the link $(j, j+1)$, $\forall 0 \leq j < n-1$. In an example Google search result for the query "paths with jumps" shown in Figure 1a, $n$=5,320,000 , $m$=10 and $v_0$ is presented as the first item in the list. Notice that in the list, the first and the fourth items may be treated by human users as the same since they refer to the same information (i.e., lattice paths with an infinite set of jumps). For the computer, however, they are different since they have different uniform resource locators (i.e., www.fpsac.org and lipn.univ-paris13.fr) Figure 1b shows the corresponding graph representation of the list, with nodes $v_0$, $v_1$, …, $v_{n-1}$ that represent the items in the list and links (0, 1), (1, 2), …, ($n$–3, $n$–2) that provide adjacency between the nodes.




(a) A Google list of items from a user-pulled query. (b) The Path $P_{5,320,000}$.

**Figure 1**. (a) An example list of items returned by Google from an information pulled by a user who queried for the topic "paths with jumps." In this example, $n$=5,320,000 while (not shown) $m$=10. (b) The corresponding graph representation of the list of items in (a).

In the graph representation of a list such as the one shown in Figure 1b, the nodes $v_0$ and $v_1$ are connected by the link (0, 1), the nodes $v_1$ and $v_2$ by the link (1, 2), and so on up to nodes $v_{n-2}$ and $v_{n-1}$ connected by the link ($n$–3, $n$–2). In general, any adjacent nodes $v_j$ and $v_{j+1}$ are connected by the undirected link $(j, j+1)$. This kind of graph is unique compared to other graphs because of the simplicity of the topology it induces. Such a special graph is called a "path" defined as follows:

**Definition 3**. A <u>path</u> $P_n$ of order $n$ is a sequence of $n$ distinct nodes $v_0$, $v_1$, …, $v_{n-1}$ connected by a sequence of ($n$–1) links (0 ,1), (1, 2), …, ($n$–3, $n$–2) such that any adjacent nodes $v_j$ and $v_{j+1}$ are connected by the link $(j, j+1)$, $\forall 0 \leq j < n-1$ [1].

The behavior of users upon encountering a list of information is to scan the presented items to quickly look for those that will satisfy their search criteria starting at $v_0$. Usually, the users are already satisfied with the first few items in the list. However, for users who have posted their search queries incorrectly, they will search deeper into the list, exhibiting a behavior that usually combines an item-by-item scan with jumps that go forward and backward into the list. An example user action with forward and backward jumps is the list of items whose visualization is shown in Figure 2b. In this example, the original path is $P_5$ (i.e., path of order $n$=5) and the user action over $P_5$ starting at $v_0$ and ending at $v_0$ is a list of visited nodes ($v_0$, $v_1$, $v_3$, $v_2$, $v_4$, $v_3$, $v_0$). Note that in the example, $v_0$ and $v_3$ are visited twice and the order of visit does not obey the adjacency imposed by the original path $P_5$ shown in Figure 2a. Such a behavior may be represented as a special graph called a "walk" defined as follows:

**Definition 4**. A <u>walk</u> $W_n$ of order $n$ is composed of $n$ distinct nodes and is a sequence of alternating nodes and links, beginning and ending with a node, where each node $v_j$ is incident to a link $(i, j)$ that precedes it and a link $(j, k)$ that follows it in the sequence, and where the nodes $v_j$ and $v_k$ that respectively precede and follow a link $(j, k)$ are the end nodes of that link [1].

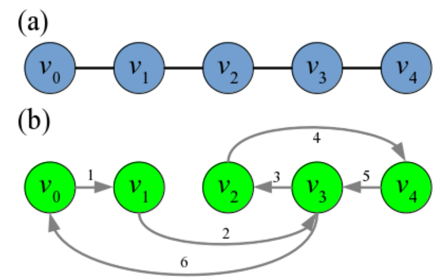



**Figure 2**. (a) An example path with $n$=5 called $P_5$. (b) A possible user behavior on $P_5$ represented as a graph with numbered arrows. The arrow represents the direction of the user's visit while the number on an arrow represent the order of the visit.

The graph shown in Figure 2b may be seen as a walk of order $n$=5 (alternatively $W_5$). It can be seen here that $W_5$ has more links than $P_5$, and the links may jump from one node to a non-adjacent node in the original $P_5$. For example, the respective nodes in the pairs $v_1$ and $v_3$, $v_2$ and $v_4$, and $v_3$ and $v_0$ were made adjacent by the jumps in $W_5$, but they are not adjacent in $P_5$. In general, $W_n$ may be generated by recording the nodes traversed by a user over $P_n$. However, the jumps in $W_n$ may not be easily modeled with $P_n$ because $P_n$ lacks the links to make the user behavior modeling possible. Thus, there is a need to define a "path with jumps" to allow the jumping behavior. Additionally, a new definition is proposed here as follows:

**Definition 5**. A walk on a path $WP_n$ of order $n$ is a walk that obey the node adjacency of a path $P_n$.

In this paper, a new graph topology based on paths is proposed. This graph topology is termed "path with jumps," denoted as $J_n$, specifically created to allow a simple walk on a path that can model user behaviors. In reality, the user's walk over $J_n$ is a time-progressed $s$ sub-walks $W_{(0)}$, $W_{(1)}$, …, $W_{(s-1)}$, where $W_{(t)}$ is a walk that was developed up to time $t \leq s$. If the nodes and links in the $W_{(i)}$'s are seen as elements of a set, then this means that $W_{(i-1)} \subset W_{(i)}$, $\forall 0 < i \leq s-1$ (where the set relational symbol $\subset$ means the set on the left is a *subset* of the set on the right). Alternatively, if the sequence of nodes and links in the $W_{(i)}$'s are seen as characters in a string, then $W_{(i-1)} \sqsubset W_{(i)}$, $\forall 0 < i \leq s-1$ (where the string relational symbol $\sqsubset$ means the string on the left is a *prefix* of the string on the right). Notice here that in general, the final walk $W_{(s-1)}$ is both a *superset* and *superstring* of all previous walks $W_{(j)}$ (i.e., $W_{(s-1)} \supset W_{(j)}$ and $W_{(s-1)} \sqsupset W_{(j)}$, $\forall 0 < j \leq s-2$).

The subset and prefix relations of any $W_{(j)}$'s to any $W_{(k)}$'s, $j < k$, are true when the underlying $J_n$ topology is non-changing (or static) at best, or under steady state at the worst. In reality, however, $J_n$ is likewise time-progressed (i.e., dynamic). Thus, this paper also presents the inherent complexities of a dynamic $J_n$, where the jumps are preserved. Finally, this paper discusses the utility of such a graph for modeling user behaviors in example online services such as Google and Facebook.

## II. THEORETICAL PRINCIPLES

This section presents the rudimentary graph principles that were used to support the definition of the proposed path with jumps. These principles are node distance, power of paths, step paths, line paths, path complement, and jump of a path, each discussed in its own subsection.

### *A. Node Distance*

Given an undirected graph $G(V, E)$, the *distance* $d_{i,j}$ between any pair of nodes $v_i$ and $v_j$ that are both in $V$ is the minimum number of links that one can go through when visiting the nodes from $v_i$ to $v_j$. If $v_i$ and $v_j$ are adjacent, then $d_{i,j}$ = 1. If $v_i$ and $v_j$ are non-adjacent, but the pairs $v_i$ and $v_k$ and $v_k$ and $v_j$ are, then one can go from $v_i$ to $v_j$ through $v_k$ and $d_{i,j}$ = 2 (Figure 3a). If one is to take the distances between all possible node pairs in $G$, then the maximum distance is called the *diameter of* $G$ [2], denoted as $\varnothing(G)$. Thus, $\varnothing(P_n)$ is $P_n$'s size $n$–1 (Figure 3b).

### *B. Power of Paths*

In a branch of mathematics called *graph theory*, the $k$th power of an undirected graph $G$, denoted as $G^k$, is another graph that has the same set of nodes $V$, but the difference is that in $G^k$, two nodes $v_i$ and $v_j$ are adjacent when their distance $d_{i,j}$ in the original graph $G$ is at most $k$ [3]. In fact, the powers of a graph $G$ has the same nomenclature as that of the exponentiation of numbers


P-ISSN 2362 – 8022 | E-ISSN 2362 – 8030 | www.apjeas.apjmr.com

such that $G^2$ is called the "square of $G$", $G^3$ is called the "cube of $G$," and so on. This general principle is used to define the power of paths as follows:

**Definition 6**. The <u>$k$th power of $P_n$</u>, denoted as $(P_n)^k$, is a graph with $V$ and that connects all pairs of nodes $v_i$ and $v_j$, where $d_{i,j} \le k$.

Figure 3c shows the $k$th powers of $P_5$, where $2 \le k \le 4$. Notice here that one can iteratively build $(P_5)^4$ from $(P_5)^3$, $(P_5)^3$ from $(P_5)^2$, and $(P_5)^2$ from $P_5$. Notice further that $(P_5)^4$ is a complete graph of order 5, denoted in the literature as $K_5$ [1]. In general, any $(P_n)^k$ can iteratively be built from $(P_n)^{k-1}$, and that $(P_n)^{n-1}$ is actually $K_n$. Since there will be no other distance greater than $\varnothing(P_n)$, then it is assumed that $(P_n)^k$ for all $k > \varnothing(P_n)$ is undefined.

*C. Step of Paths*

The <u>$k$th step of a path</u> $P_n$, denoted as $(P_n)^{[k]}$, is a graph that has the same $V$ as $P_n$, but its link set $E$ is the link set of $(P_n)^k$ less the link set of $P_n$ (Equation 1). Figure 3d shows the $k$th steps of $P_5$, for all $2 \le k \le 4$. Notice that the 2-step of $P_5$ resulted in two separate paths $P_2$ with nodes $v_1$ and $v_3$, and $P_3$ with nodes $v_0$, $v_2$, and $v_4$. It is easy to see that in general, a 2-step of $P_n$ results in $P_{\lfloor n/2 \rfloor}$ with odd-numbered nodes from $P_n$ and $P_{\lceil n/2 \rceil}$ with even-numbered nodes from $P_n$. A $k$-step of $P_n$ for $k > 2$ results in a connected graph with cycles. For example, $(P_5)^{[3]}$ in Figure 3c is a connected graph with cycles.

$$E((P_n)^{[k]}) = (\, E((P_n)^k) \cap E(P_n)\, )' \qquad (1)$$

*D. Line Paths*

The <u>line path</u> $\mathscr{L}(P_n)$ of a path $P_n$ is another path that represents the adjacencies between the links of $P_n$ [4]. It is easy to see that $\mathscr{L}(P_n)$ is of order $n - 1$, and that the line of $\mathscr{L}(P_n)$, denoted as $\mathscr{L}(\mathscr{L}(P_n))$ or its shorter alternate $\mathscr{L}^2(P_n)$, is a path $P_{n-2}$ whose node set is the same as that of $P_n$ less the two endpoints $v_0$ and $v_{n-1}$ of $P_n$. In general, $\mathscr{L}^k(P_n)$ is $P_{n-k}$, where $k < n$; if $k$ is odd then $V(\mathscr{L}^k(P_n)) \subset E(P_n)$, while if $k$ is even then $V(\mathscr{L}^k(P_n)) = (\, V(P_n) \cap \{v_0, v_1, ..., v_{k/2-1}, ..., v_{n-k/2}, ..., v_{n-2}, v_{n-1}\}\, )'$, which is $V(P_n)$ less the $k$ end nodes of $P_n$. Figure 3e shows the corresponding line path of $P_5$.

*E. Path Complement*

The <u>complement of a path</u> $P_n$, denoted as $(P_n)^c$, is a graph such that the node set $V$ of $P_n$ is also the node set of $(P_n)^c$ but the link set of $(P_n)^c$ is the link set of $K_n$ less the link set of $P_n$. [1]. It is easy to see that $(P_n)^c$ is nothing but $(P_n)^{[n-1]}$. Figure 3d also shows the path complement of $P_5$.

*F. Jump of a Path*

The <u>jump of a path</u> $P_n$, denoted as $J(P_n)$, is a graph that is $(\mathscr{L}(P_n))^c$ [5]. Note here that the node set of $J(P_n)$ is defined over the link set of $P_n$. Figure 3f shows $J(P_5)$.

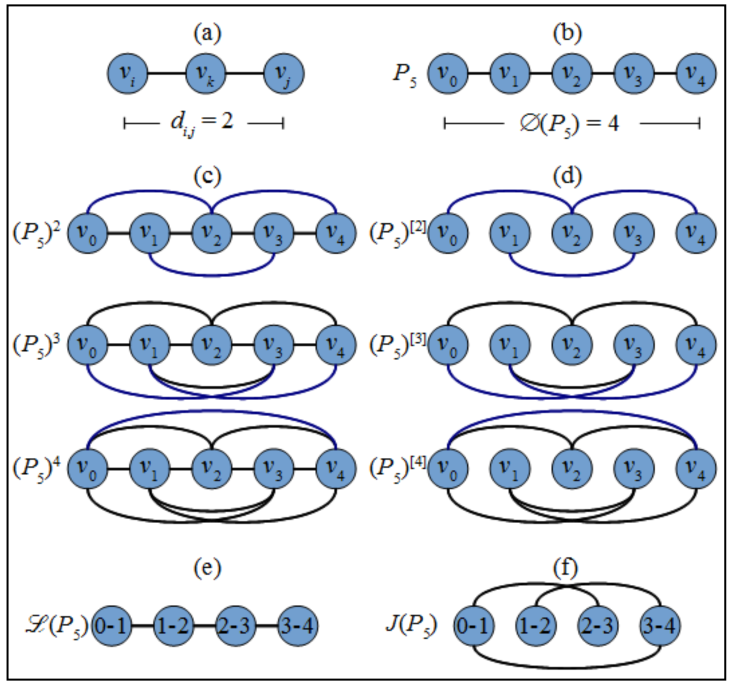



**Figure 3**. (a) Node distance $d_{i,j}$ between two nodes $v_i$ and $v_j$ with node $v_k$ between them. An example path $P_5$ showing: (b) its diameter $\varnothing(P_5)$=4; (c) its three powers, namely $(P_5)^2$, $(P_5)^3$, and $(P_5)^4$; (d) its three steps $(P_5)^{[2]}$, $(P_5)^{[3]}$, and $(P_5)^{[4]}$; (e) its line $\mathcal{L}(P_5)$; and (f) its jump $J(P_5)$. Notice that $K_5 = (P_5)^4$ and $(P_5)^c = (P_5)^{[4]}$.

## III. PATHS WITH JUMPS

In this section, the development of a new graph called path with jumps is discussed. Prior to its definition, two new graph definitions based on $P_n$ are proposed. These graphs are respectively called the *jumpsteps of a path*, and the *combined jumpsteps of a path*, defined in their own subsections.

*A. The Jumpsteps of a Path*

A new proposed graph called the $k$th jumpstep of a path $P_n$ is defined as follows:

**Definition 7**. The <u>$k$th jumpstep of $P_n$</u>, denoted as $JS_k(P_n)$, is a graph with the same node set $V$ as $P_n$ but its link set is defined as $E(JS_k(P_n)) = (E((P_n)^k) \cap E((P_n)^{k-1})\, )'$.

Figure 4a shows the $k$th jumpsteps of $P_5$, for all $2 \leq k \leq 4$. It is easy to see that alternatively, $E(JS_k(P_n)) = (E((P_n)^{[k]}) \cap E((P_n)^{[k-1]}))'$. This is trivial because the difference between $E((P_n)^k)$ and $E((P_n)^{[k]})$ is $E(P_n)$. The 2nd jumpstep of $P_n$ is also $(P_n)^{[2]}$, which provide the basis for $JS_k(P_n)$. Just like $(P_n)^{[2]}$, $JS_2(P_n)$ results into two disjoint $P_{\lfloor n/2 \rfloor}$ and $P_{\lceil n/2 \rceil}$. $JS_3(P_n)$ results into three disjoints paths namely two $P_{\lceil n/3 \rceil}$'s and one $P_{\lfloor n/3 \rfloor}$. In general, $JS_k(P_n)$ results into $k$ disjoint $P_{n/k}$'s. In an extreme case, $JS_{n-1}(P_n)$ results into one $P_2$ with nodes $v_0$ and $v_{n-1}$ connected to each other, and $(k-1)$ $P_1$'s.

*B. The Combined Jumpsteps of a Path*

Each of the individual jumpsteps of any $P_n$ may be combined either to recreate the other graphs discussed above or to form a new graph. For example, the combined 2nd and 3rd jumpsteps of $P_n$ is a recreation of the 3rd-step of $P_n$, while the combined 2nd and 4th jumpsteps of $P_n$ is a whole new graph. Intuitively, the combined 2nd, 3rd, and up to $(n-1)$th jumpsteps of $P_n$ is $(P_n)^{[n-1]}$. Throughout this paper, the combined jumpsteps of $P_n$ is denoted as $JS_{\{B\}}(P_n)$, where $B$ is a non-negative integer whose binary representation is $n$-long, wherein its least significant bit is always 0 and its $k$th bit is 1 if $JS_k(P_n)$ is included in the combination, $\forall 1 < k \leq n$. For example, if the 2nd and 4th jumpsteps of $P_n$ is to be combined, then the 2nd and 4th bits of the $n$-long binary representation of $B$ is set to 1, while the other bits are set to zero resulting to $1010_2$ which is $10_{10}$. The combined jumpsteps is denoted $JS_{\{10\}}(P_n)$. If for example the 4th and 5th jumpsteps are to be combined, then the combination is denoted $JS_{\{24\}}(P_n)$. Figure 4b shows $JS_{\{10\}}(P_6)$ and $JS_{\{24\}}(P_6)$ as examples.

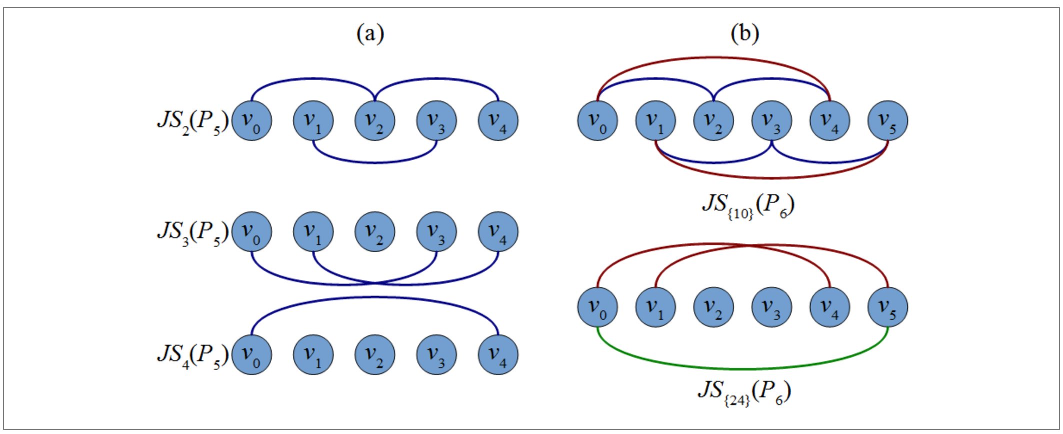



**Figure 4**. (a) The $k$th jumpsteps of $P_5$: $JS_2(P_5)$, $JS_3(P_5)$, and $JS_4(P_5)$; and (b) the combined 2nd and 4th jumpsteps of $P_6$, and 4th and 5th jumpsteps of $P_6$.

*C. Paths with Jumps*

The new graph presented here, called a path with jumps, is arguably different from $J(P_n)$. For one, $J(P_n)$'s node set is defined over $E(P_n)$, and the path with jumps is by no means bigger in degree and in size than $J(P_n)$. A path with jumps is defined as follows:

**Definition 8**. A path of order $n$ with $m$ jumps, denoted as $J_{n,m,k,d}$, is a sequence of $m$ $P_k$-separated $JS_{\{B\}}(P_d)$'s, where $d = n/m - (m-1)\lfloor k/2 \rfloor + 1$ and $k \geq 2$.

To visualize this new graph, Figure 5a shows a path of order $n$=8 with $m$=2 jumps. Here $k$=2, so the jump separator is $P_2$, while the two jumps $JS_{\{6\}}(P_4)$ have 4 nodes and are both $K_4$. Note that $P_2$ shares its only two nodes with both $JS_{\{6\}}(P_4)$'s. This example is denoted $J_{8,2,2,4}$ and is also called a dumbbell graph in graph theory [6]. One may see that $J_{n,m,k,d}$ is a generalization of the dumbbell graph for even $m > 2$. This is because for an even $m > 2$, one can see $m/2$ dumbbell graphs that are separated by $P_k$'s (Figure 5b).

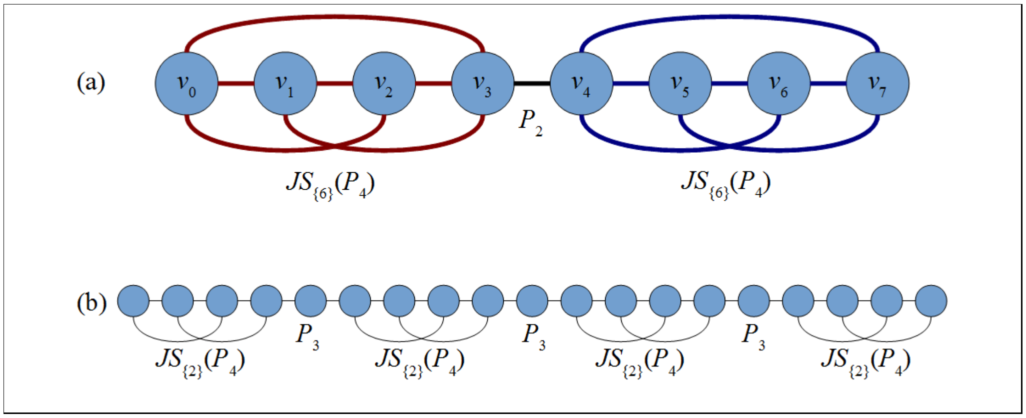



**Figure 5**. (a) An example path with jumps $J_{8,2,2,4}$, and (b) another example $J_{19,4,3,4}$ showing two dumbbell graphs joined by a $P_3$, where the dumbbell graphs themselves are two $P_3$-separated $JS_{\{2\}}(P_4)$'s. Note that a $P_3$ shares its two end nodes with the two $JS_{\{2\}}(P_4)$'s.

## IV. TOPOLOGY-PRESERVING DYNAMIC PATH WITH JUMPS

Figure 6a shows a snapshot of the 7th page of a Google query result for the information pull "Katy Perry." Note that Katy Perry is the world's most popular celebrity based on the number of Twitter followers [7]. It is easy to infer from the figure that the list has $n$=329,000,000. The default setting for a Google search has each page lists $m$=10 items. The snapshot is already (partly) annotated with nodes indexed from 141 through 149. The 329M-item Google list can be modeled as $J_{n,m,2,10}$ over which will allow a simple walk by users.

Figure 6b, on the other hand, shows the snapshot of the same 7th page after some time with a noticeable change for item 145. Although not shown in the figure, the previous item 145 was moved as item 146. This means that the new item 145 was just inserted to displace the old item 145. This also means that the previous $n$th item is now the $(n + 1)$th item, for all $n$ > 145. Thus, the model is now $J_{n+1,m,2,10}$ but the topology is still the same, particularly for a very large $n$. In general, an insertion of a node into, a deletion of a node from, or a movement of a node within the path makes the path with jumps dynamic, yet its topology is preserved.

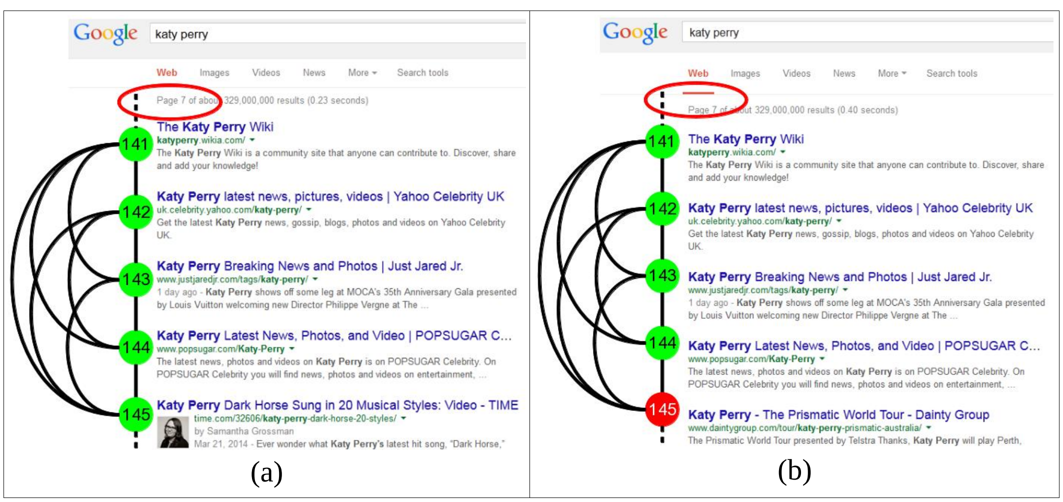



**Figure 6**. (a) An example 7th page of a list of items returned by Google from an information pulled by a user who queried for the topic "Katy Perry." (b) The same example 7th page of the "Katy Perry" list but with an inserted item.

Since $n$ is very large and $m$ is very small, the generated path in real-world modeling results in a very long path. To simplify the visualization, the graphical representation used in power graph analysis in the domain of computational biology is utilized here [8]. The diagrammatic representation of the three primitive motifs, the biclique, the clique, and the star, used in power graph analysis is shown in Figure 7(a). Notice that the biclique is nothing but a complete bipartite graph. The power graph representation of $J_{n,m,2,10}$, where each $JS_{\{B\}}(P_{10})$ is a $K_{10}$, is shown in Figure 7(b).

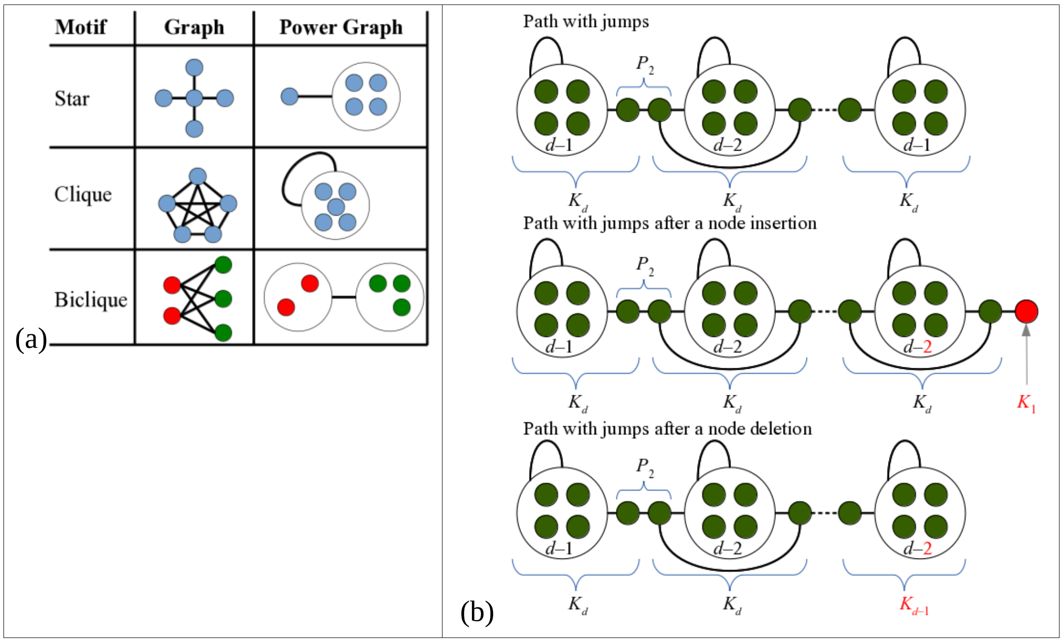



**Figure 7**. (a) The primitive motifs of a power graph analysis; and (b) The power graph representation of $J_{n,m,2,10}$, where each $JS_{\{B\}}(P_{10})$ is a $K_{10}$., before and after a node insertion or a node deletion.

Two questions need to be asked when analyzing the dynamics of $J_{n,m,k,d}$: (1) what is the cost of node insertion?; and (b) what is the cost of node deletion? Either change can affect the nodes in $P_k$. In cases of $k$=2 and node insertion, for example, the left end node of any $P_2$ along the path with jumps, formerly a node of the clique $K_d$ at the left, will become the right end node of $P_2$ after insertion and become a node of the clique at the right. The node moves to the right, severs $d – 1$ links from the left $K_d$, and connects to $d – 1$ links to its right. In case of deletion, the reverse happens: The node moves to the left, severs $d – 1$ links from the right $K_d$, and connects to $d – 1$ links to its left.

The worst case will happen if the start node $v_0$ gets deleted or replaced by a new node. Since there are $(m – 1)$ $P_2$'s in the whole $J_{n,m,k,d}$, then the cost $C$ of the change (either node insertion or node deletion) is (using some equalities defined in Definition 8):

$$\begin{aligned} C &= 2\,(m-1)(d-1) \\ &= 2\,(m-1)(n/m-(m-1)\lfloor k/2 \rfloor) \\ &\equiv O(n) \end{aligned} \quad (2)$$

## V. APPLICATIONS

The utility of $J_{n,m,k,d}$ to provide a graph for walking on a path as an easy model of a user's behavior traversing the list of items in the Google search result is underscored here. In fact, the development of the path with jumps was aided by the Google example. In reality, however, the dynamism in the Google example only takes into consideration those changes with node insertion. Examples of dynamic paths with jumps to model the results of other web services, which not only include node insertion but node deletion and node movement as well, are those of The Pirate Bay and Facebook. Other web services, of course, can also be modeled but these two (aside from Google) were selected because of their popularity among the Internet users.

### *A. Node Deletions in TPB*

The Pirate Bay (www.thepiratebay.se), popularly known as TPB, is a website the provides magnet links [9] to torrent files used in a content distribution network called the BitTorrent network [10]. When a user pulls information from this service via a query for a torrent file, the service returns a list of *n* items in *m* pages, similar to how Google returns the information pull. However, aside from item insertion, the dynamic list also experiences item deletion. Since Google only experiences node insertion, then those nodes that are located to the left of the insertion will have indeces that are higher than the previous indeces before the insertion. In constrast to Google, the node index of the path with jumps representation of the list for TPB may either go up and down. The dynamic of TPB is much more difficult to model because of the unpredictability of when a node will be inserted into or deleted from the list.

### *B. Node Movements in FB*

Facebook (FB) experiences both node insertion and node movement in its path with jumps representation of the list of posts. One of the most complained difficulty in traversing this list is the automatic movement of the post to the top of the list while the post is being read. This provides frustrations to users who would scroll to the top again just to continue reading the moved post. Modeling movement of items with the list is rather easy because a node movement along the path is not a basic change compared to node insertion and deletion. Node insertion and deletion are both fundamental changes. Node movement can be modeled as a simultaneous deletion of a node $v_k$ with an insertion of the same node $v_k$ at the start of the path.

## VI. SUMMARY AND CONCLUSION

This paper introduces a new graph definition called path with jumps, denoted as $J_{n,m,k,d}$. This new graph provides an easy modeling for user's behavior utilizing web services that provide a list of items that are either pulled by the user or pushed by the service to the users. The dynamics of $J_{n,m,k,d}$ was discussed, which include node insertion, node deletion, and node movement. Node deletion usually happens in the dynamic list of Google. Node insertion and deletion happen in the dynamic list of torrent files in TPB, while node insertion and node movement happen in the dynamic list of posts in FB. Both node insertion and deletion were considered fundamental changes in $J_{n,m,k,d}$, while node movement is not a fundamental change. However, node movement can be modeled as a concurrent node deletion and node insertion at the top of the list. Whether the change is node insertion or deletion, the cost of dynamism in $J_{n,m,k,d}$ is always $O(n)$, where *n* is the number of items in the list.

The following were contributed in this work: (1) defined walk on a path (Definition 5); (2) proposed a new graph called the the *k*th jumpstep of a path (Definition 7); (3) proposed a new graph called the combined jumpsteps of a path; (4) proposed a new graph called path with jumps (Definition 8); (5) discussed the topology-preserving dynamic path with jumps (Section IV); and (6) provided example web services whose users' behavior may be modeled by these proposed graphs.

## ACKNOWLEDGEMENTS

This research effort is funded by the Institute of Computer Science (ICS) of the University of the Philippines Los Baños (UPLB) under the research programs *Structural Characterization and Temporal Dynamics of Various Natural, Social and Artificial Networks in the Philippines* and *Evaluating Approaches for Modeling and Simulating the Structure and Dynamics of Artificial Societies*.

## REFERENCES

[1] F. Harary. (1994). **Graph Theory, Advanced Book Program Series**. Westview Press: Boulder, Colorado, pp 284 (ISBN: 0201410338).

[2] J.M. Harris, J.L. Hirst and M.J. Mossinghoff. (2000). **Combinatorics and Graph Theory, Springer Undergraduate Texts in Mathematics and Technology Series**. Springer-Verlag: New York, pp 248 (ISBN: 0387987363).

[3] J.A. Bondy and U.S.R. Murty. (2008). **Graph Theory, Graduate Texts in Mathematics Series 244**. Springer: New York, pp 654 (ISBN: 1846289696).

[4] H. Whitney. (1932). *Congruent graphs and the connectivity of graphs*. **American Journal of Mathematics** 54(1):150–168 ( DOI: 10.2307/2371086).

[5] G. Chartrand, H. Hevia, E.B. Jarrett and M. Schultz. (1997). *Subgraph distances in graphs defined by edge transfers*. **Discrete Mathematics** 170(1–3):63–79 ( DOI: 10.1016/0012-365X(95)00357-3).

[6] J. Wang, F. Belardo, Q. Huang and E.M. Li Marzi. (2010). *Spectral Characterizations of Dumbbell Graphs*. **The Electronic Journal of Combinatorics** 17:R42 (ISSN: 1077-8926).

[7] J.P. Pabico. (2014). *An Empirical Investigation of Preferential Attachment Among Influential Members of a Large Artificial Community*. **Asia Pacific Journal of Multidisciplinary Research** 2(1):178–188 (ISSN: 2350-7756).

[8] L. Royer, M. Reimann, B. Andreopoulos and M. Schroeder. (2008). *Unraveling protein networks with power graph analysis*. **PloS Computational Biology** 4(7):e1000108 (DOI: 10.1371/journal.pcbi.1000108).

[9] J. Chapweske. (2001). *HTTP Extensions for a Content-addressable Web*. W3C's www-talk (http://list.w3.org).

[10] I.C.B. Flores and J.P. Pabico. (2010). *When leechers are left to tend for themselves: Performance of orphaned file sharing networks*. In H.N. Adorna and R.P. Saldaña (eds.) **Proceedings of the 10th Philippine Computing Science Congress (PCSC 2010)**, Ateneo de Davao University, Davao City. pp. 212-218 (ISSN: 1908-1146).