ارائه یک مدل جدید از برپایی حمله سایبیل در شبکه‌های حسگر بی‌سیم مبتنی بر خوشه‌بندی و راه‌کار مقابله با آن

مجتبی جمشیدی[۱]، احسان زنگنه[۲]، مهدی اثنی عشری[۳]، محمدرضا میبدی[۴]

[۱] آموزشکده فنی و حرفه‌ای سما، دانشگاه آزاد اسلامی، واحد کرمانشاه، کرمانشاه، ایران jamshidi.mojtaba@gmail.com

[۲] گروه کامپیوتر، موسسه آموزش عالی جهاد دانشگاهی کرمانشاه، کرمانشاه، ایران ehsan.zangneh@yahoo.com

[۳] پژوهشکده فناوری اطلاعات، پژوهشگاه فضای مجازی، تهران، ایران esnaashari@itrc.ac.ir

[۴] دانشکده کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران mmeybodi@aut.ac.ir

چکیده: یکی از حمله‌های خطرناک شناخته شده علیه شبکه‌های حسگر بی‌سیم، حمله سایبیل است. در این حمله، گره بدخواه همزمان چندین شناسه جعلی از خود منتشر می‌کند که این سبب می‌شود پروتکل‌های مسیریابی و عملیاتی نظیر رأی‌گیری و تجمیع داده‌ها تا حد زیادی تحت تأثیر قرار گیرند. در این مقاله، ابتدا یک مدل جدید از برپایی حمله سایبیل در شبکه‌های حسگر مبتنی بر خوشه‌بندی مطرح می‌گردد. سپس یک الگوریتم، مبتنی بر مشخصه قدرت سیگنال دریافتی (*RSSI*) و مکان‌یابی به کمک سه نقطه جهت مقابله با این مدل جدید از حمله پیشنهاد می‌گردد. الگوریتم پیشنهادی شبیه‌سازی شده و با انجام یک‌سری آزمایش‌ها، کارایی آن از نقطه نظرهای نرخ تشخیص درست، نرخ تشخیص غلط و سربار ارتباطات ارزیابی گردیده است. نتیجه آزمایش‌ها نشان داد الگوریتم پیشنهادی با تحمیل سربار ارتباطی اندکی به شبکه، قادر به شناسایی ۹۹٫۸% گره‌های سایبیل و نرخ تشخیص غلط ۰٫۰۸% است (در حالت میانگین).

کلمات کلیدی: شبکه حسگر بی‌سیم، حمله سایبیل، خوشه‌بندی.

۱- مقدمه

یک شبکه حسگر حاوی تعداد زیادی گره حسگر منابع محدود (از نظر انرژی، حافظه، برد رادیویی، پردازش و ...) است. خوشه‌بندی یکی از مکانیزم‌های مقیاس‌پذیری و موثر در کاهش ترافیک شبکه و انرژی مصرفی است. در این مکانیزم، شبکه به چند قسمت موسوم به خوشه تقسیم می‌شود و در هر خوشه یکی از گره‌ها با ایفای نقش "سرخوشه"، وظیفه تجمیع داده‌های جمع‌آوری شده توسط اعضای خوشه و در صورت لزوم ارسال داده‌ها به ایستگاه پایه را بر عهده می‌گیرد[1] [2]. یکی از چالش‌های پیش روی شبکه‌های حسگر خوشه‌بندی شده، مسئله امنیت است. حمله سایبیل (*Sybil*) یکی از خطرناک‌ترین حمله‌های تأثیرگذار در لایه مسیریابی است که امکان برپائی آن در شبکه‌های حسگر مبتنی بر خوشه‌بندی نیز وجود دارد. در این حمله، گره بدخواه دشمن همزمان چندین شناسه جعلی از خود منتشر می‌کند. این امر سبب می‌شود گره‌های نرمال در همسایگی گره بدخواه فریب خورده و به اشتباه گمان کنند همسایه‌های زیادی دارند. از این‌رو، گره بدخواه ترافیک زیادی به خود جذب نموده و به‌طور چشمگیری پروتکل‌های مسیریابی را مختل می‌کند و نیز می‌تواند بر عملیاتی نظیر تجمیع داده‌ها، تخصیص منابع و رأی‌گیری تأثیر گذارد و یا حتی داده‌های غلط در شبکه تزریق کند [3] [4] [5].

در این مقاله، یک مدل جدید از راه‌اندازی حمله سایبیل در شبکه‌های حسگر بی‌سیم خوشه‌بندی شده پیشنهاد می‌شود. هم‌چنین یک الگوریتم جدید جهت مقابله با این مدل پیشنهادی از حمله سایبیل مطرح می‌گردد.

ادامه این مقاله بدین ترتیب سازماندهی می‌شود. در بخش ۲، کارهای گذشته آمده است. فرضیات سیستم و مدل حمله در بخش ۳ شرح داده می‌شود. در بخش ۴ الگوریتم پیشنهادی و در بخش ۵ ارزیابی کارایی و نتایج شبیه‌سازی ارائه می‌شود. بخش آخر نیز به نتیجه‌گیری می‌پردازد.

۲- کارهای گذشته

حمله سایبیل نخستین بار توسط Douceur در [3] معرفی شد و اشاره شد که شبکه‌های نظیر به نظیر در برابر این حمله آسیب‌پذیر هستند. Karlof در [4] بیان کرد این حمله می‌تواند بر پروتکل‌های مسیریابی شبکه‌های حسگر تأثیرگذار باشد. Newsome و همکارانش در [5] برای اولین بار به‌طور سیستماتیک به بررسی و تحلیل حمله سایبیل برای شبکه‌های حسگر بی‌سیم پرداختند و راه‌کارهایی نظیر تست منبع رادیویی و پیش‌توزیع تصادفی کلیدها برای مقابله با این حمله مطرح گردید. در [6] و [7] الگوریتم‌هایی مبتنی بر RSSI[۱] جهت شناسایی گره‌های سایبیل در شبکه‌های حسگر مبتنی بر پروتکل مسیریابی LEACH ارائه شده است که مدل حمله سایبیل در نظر گرفته شده در آنها، چنین است که گره بدخواه (گره‌های سایبیل) به عنوان سرخوشه عمل می‌کند. در [8] نیز یک الگوریتم جهت شناسایی گره‌های سایبیل در شبکه‌های حسگر متحرک که مبتنی بر الگوریتم "خوشه‌بندی کمترین *ID*" [9] هستند ارائه شده است. در مرجع [11] از مکانیزم تعیین مکان ارائه شده در [10] بهره گرفته و یک الگوریتم جهت شناسایی گره‌های سایبیل مطرح شده است. در [12] یک الگوریتم توزیعی به منظور شناسایی گره‌های سایبیل ارائه شده است که فقط از اطلاعات مربوط به تعداد همسایه‌ها استفاده می‌کند تا گره‌های سایبیل را شناسایی کند. در [13-20] نیز الگوریتم‌های دیگری جهت مقابله با این حمله در شبکه‌های حسگر غیرخوشه‌بندی شده ارائه شده است.

۳- فرضیات سیستم و مدل حمله

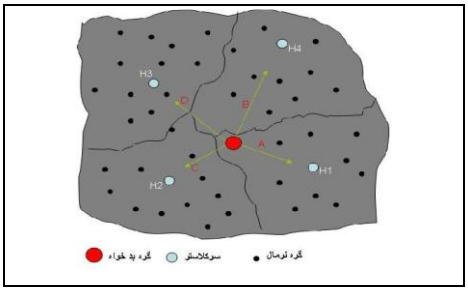
۳-۱- فرضیات سیستم

شبکه حسگر حاوی N گره حسگر است که به طور تصادفی در یک ناحیه دوبعدی توزیع می‌شوند. گره‌ها ساکن بوده و از موقعیت مکانی خود آگاه نیستند. شبکه همگن (همه گره‌ها امکانات سخت‌افزاری و نرم‌افزاری برابری دارند) بوده و هر گره یک شناسه یکتا دارد. هم‌چنین، فرض می‌شود برد رادیویی (r) همه گره‌ها یکسان و به حدی است که کل محیط عملیاتی را پوشش می‌دهد. شبکه، خوشه‌بندی شده است و تعداد خوشه‌ها C_{opt} می‌باشد.

۳-۲- مدل حمله

در [8] و [9] یک مدل از راه‌اندازی حمله سایبیل در شبکه‌های حسگر خوشه‌بندی شده مطرح گردیده است. در این مدل، گره بدخواه تلاش می‌کند به عنوان سرخوشه انتخاب شود تا تأثیر زیادی بر عملکرد شبکه گذارد. ولی از آنجا که گره‌های سرخوشه معمولا در ارتباط مستقیم با ایستگاه پایه هستند و نقش مهمی در شبکه ایفا می‌کنند لذا رفتارشان به شدت توسط ایستگاه پایه کنترل می‌شوند. بنابر این، در حمله سایبیل، اگر گره بدخواه نقش سرخوشه داشته باشد و یا تلاشی در این جهت انجام دهد، به سرعت توسط ایستگاه پایه شناسایی می‌شود. چراکه، گره ایستگاه پایه از لحاظ منابع نامحدود بوده و می‌تواند حاوی الگوریتم‌های امنیتی پیچیده و مقاوم باشد. هم‌چنین، ایستگاه پایه معمولا از تمام اطلاعات شبکه، نظیر تعداد گره‌ها، شناسه دیگر گره‌ها، کلیدهای رمزنگاری و ... آگاه است.

در این‌جا، یک مدل جدید از راه‌اندازی حمله سایبیل در شبکه‌های حسگر خوشه‌بندی شده ارائه می‌گردد. مدل حمله پیشنهادی بدین صورت است که در فاز شکل‌گیری خوشه‌ها، پس از این‌که گره‌های سرخوشه (که تعداد آن‌ها C_{opt} است) پیغام کاندیداتوری خود را در شبکه منتشر می‌کنند، گره بدخواه با دریافت این پیغام‌های کاندیداتوری، S شناسه سایبیل تولید کرده و با هر شناسه به یک سرخوشه می‌پیوندد. در این مدل حمله، گره بدخواه تلاشی برای سرخوشه شدن انجام نمی‌دهد بلکه تلاش می‌کند با هر شناسه سایبیل خود عضو یک خوشه خاص شود. از این‌رو گره بدخواه همزمان عضو چندین و یا حتی همه‌ی خوشه‌ها خواهد شد. در نتیجه عملیات و داده‌های چندین خوشه و یا حتی تمام خوشه‌های شبکه را تحت تأثیر قرار می‌دهد. شکل (۲)، یک مثال از راه‌اندازی این مدل از حمله سایبیل را نشان می‌دهد.



شکل ۱: یک نمونه از راه‌اندازی مدل جدید حمله سایبیل

در این مثال، گره بدخواه با شناسه A عضو سرخوشه $H1$، با شناسه B عضو $H4$، با شناسه C عضو $H2$ و با شناسه D عضو $H3$ است. توجه شود که گره بدخواه با توان‌های مختلف پیغام *join* را به سرخوشه‌ها ارسال می‌کند. هرچه سرخوشه به گره بدخواه دورتر باشد، با توان بیشتری پیغام *join* را ارسال می‌کند تا به دست آن سرخوشه برسد.

مدل حمله پیشنهادی، برخلاف مدل مطرح شده در [8] و [9]، این سبب می‌شود، ۱- گره بدخواه به راحتی توسط ایستگاه پایه شناسایی نشود و ۲- گره بدخواه همزمان بر روی چند خوشه تأثیر گذارد (به عنوان مثال، با تزریق داده‌های نادرست). این مدل از راه‌اندازی حمله سایبیل تا به حال در هیچ تحقیقی مطرح نشده است، اما با توجه به تأثیری زیادی که می‌تواند بر روی بخش عظیمی از شبکه بگذارد، در اینجا معرفی شده و در بخش بعدی یک راه‌کار جهت مقابله با آن پیشنهاد می‌گردد.

۴- الگوریتم پیشنهادی

ایده اصلی الگوریتم پیشنهادی برگرفته از این موضوع است که همه گره‌های سایبیل در یک مکان از شبکه قرار دارند چراکه همه آن‌ها مربوط به یک سخت‌افزار یکتا (گره بدخواه) هستند. همان‌طور که پیش‌تر گفته شد، در فاز شکل‌گیری خوشه‌ها، گره‌های سرخوشه یک پیغام فراگیر در کل محیط شبکه منتشر می‌کنند. هر یک از دیگر گره‌ها با دریافت این پیغام‌ها، مطابق یک‌سری معیارها (نظیر فاصله تا گره‌های سرخوشه، میزان انرژی آن‌ها) یکی از سرخوشه‌ها را انتخاب می‌کنند تا عضو خوشه‌ی آن شوند. سپس هر گره یک پیغام *join* به سرخوشه انتخابی خود ارسال می‌کند. هر گره سرخوشه، با دریافت پیغام‌های *join*، شناسه گره ارسال کننده پیغام *join* و فاصله تقریبی با آن را در جدولی به نام *member_table* (شکل ۲) در حافظه خود ذخیره می‌کند. گره سرخوشه، شناسه هر عضو خود را در فیلد *member_ID* و فاصله تخمینی تا آن را در فیلد *est_distance* ذخیره می‌کند. گره‌های سرخوشه از شاخص قدرت سیگنال دریافتی (*RSSI*) جهت تخمین فاصله گره‌های عضو با خود استفاده می‌کنند. با فرض این‌که گره i یک سیگنال رادیویی با توان P_0 برای گره j ارسال کند، در این صورت، توان سیگنال دریافتی در گیرنده از رابطه (۱) بدست می‌آید.

$$R_{ij} = \frac{P_0 K}{\left(\hat{d}_{ij}\right)^{\alpha}} \qquad (۱)$$

که در این رابطه، R_{ij} توان سیگنال دریافتی در گیرنده (یا همان شاخص قدرت سیگنال دریافتی در گیرنده)، K یک عدد ثابت، d_{ij} فاصله اقلیدوسی بین دو گره i و j، و α فاکتور تضعیف سیگنال است. $\alpha \in \{2,4\}$ می‌باشد و توجه شود که به دلیل خطاهای اندازه‌گیری، فاصله دو گره i و j (d_{ij}) فقط می‌تواند تخمین زده شود ($\hat{d}_{ij}$) [13].

member_ID	*est_distance*
a	12
...	...

شکل ۲: جدول *member_table* گره‌های سرخوشه

توجه شود که عمل شکل‌گیری خوشه‌ها ممکن است بارها در طول حیات شبکه صورت پذیرد. از این‌رو، در هر دور خوشه‌بندی، سرخوشه‌ها باید فاصله گره‌های عضو تا خود را تخمین زنند. پس از هر دور خوشه‌بندی و تخمین فاصله‌ها بین سرخوشه‌ها و گره‌های عضو، گره‌های سرخوشه با همکاری یکدیگر الگوریتم پیشنهادی جهت تشخیص گره‌های سایبیل را اجرا می‌کنند. الگوریتم پیشنهادی از دو فاز تشکیل شده است:

۴-۱- فاز اول الگوریتم پیشنهادی (فاز پویش)

در فاز اول، هر گره سرخوشه CH_i، ابتدا میانگین فاصله تا عضوهای خود را محاسبه کرده و در یک فیلد به نام *avg_distance* در حافظه خود ذخیره می‌کند. سپس جدول *member_table* خود را پیمایش کرده و گره‌های عضوی که فاصله تخمینی تا آن‌ها بیشتر از *avg_distance* باشد، به عنوان عضوهای مشکوک در لیستی به نام SL[۲] ثبت می‌کند. چراکه گره بدخواه اگر همزمان عضو S خوشه باشد، در این صورت به احتمال p حداقل در یکی از خوشه‌ها جزء عضوهایی خواهد بود که فاصله بیشتر از *avg_distance* با سرخوشه‌ی خود دارد. هرچه S بزرگتر باشد، p نیز افزایش می‌یابد. SL سرخوشه CH_i را با SL_{CH_i} نشان می‌دهیم. بنابر این، در فاز پویش، هر گره سرخوشه، SL خود را تهیه می‌کند تا در فاز دوم به کمک دیگر سرخوشه‌ها اقدام به شناسایی گره‌های بدخواه (سایبیل) احتمالی کند. فاز پویش به‌طور همزمان توسط تمام سرخوشه‌ها اجرا می‌شود.

۴-۲- فاز دوم الگوریتم پیشنهادی (فاز آزمون)

در این فاز، هر گره سرخوشه CH_i، عضو $u \in SL_{CH_i}$ را که بیشترین فاصله تا آن را دارد به عنوان نمونه آزمایشی (ES[۳]) انتخاب می‌کند. نمونه آزمایشی سرخوشه‌ی CH_i را با ES_{CH_i} نشان می‌دهیم. سپس، هر سرخوشه، در یک بسته، نمونه آزمایشی خود (شامل شناسه گره عضو و فاصله تخمینی تا آن) را برای دیگر سرخوشه‌ها ارسال می‌کند. از این‌رو، هر گره سرخوشه، نمونه آزمایشی تمام دیگر سرخوشه‌ها را خواهد داشت. هر گره سرخوشه، نمونه آزمایشی خود و دیگر سرخوشه‌ها را در لیستی به نام *ES_List* در حافظه خود ذخیره می‌کند. هر عضو از این لیست *ES_List* در واقع نمونه آزمایشی مربوط به یک سرخوشه خاص را نشان می‌دهد. توجه شود که *ES_List*ها در تمام سرخوشه‌ها مقادیر یکسانی دارند. حال، گره سرخوشه CH_iی که بیشترین فاصله تا نمونه آزمایشی خودش، یعنی ES_{CH_i}، را داشته باشد به عنوان سرخوشه آغاز کننده روال تشخیص گره بدخواه انتخاب می‌شود. اگر حالتی رخ دهد که بیش از یک سرخوشه، نمونه آزمایشی آنها بیشترین فاصله را داشته باشد در این صورت سرخوشه با کوچکترین شناسه به عنوان آغازگر روال تشخیص انتخاب می‌شود. به این دلیل اولویت را به ES با بیشترین فاصله می‌دهیم زیرا احتمال بدخواه بودن این گره (با توجه به مدل حمله پیشنهادی) بیشتر است.

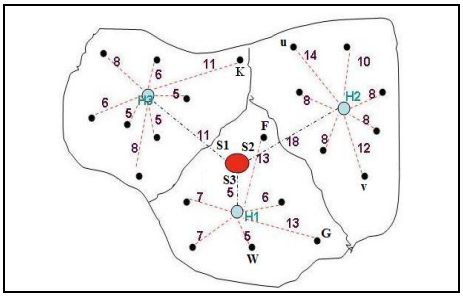
سرخوشه CH_i پس از این‌که به عنوان آغازگر روال تشخیص انتخاب شد، یک پیغام برای برای نمونه آزمایشی خود، یعنی ES_{CH_i} ارسال می‌کند. در این پیغام به نمونه آزمایشی دستور داده می‌شود یک پیغام (با همان قدرت سیگنالی که پیغام *join* را پیش‌تر در فاز خوشه‌بندی به سرخوشه CH_i ارسال کرده بود) منتشر کند (به عبارت دیگر، *broadcast* کند) تا این پیغام به دست دیگر گره‌های سرخوشه برسد. البته توجه شود که این پیغام منتشر شده توسط نمونه آزمایشی فقط به دست سرخوشه‌هایی می‌رسد که فاصله آن‌ها تا نمونه آزمایشی، کوچکتر یا مساوی فاصله نمونه آزمایشی تا سرخوشه CH_i است. چراکه گره انتخاب شده به عنوان نمونه آزمایشی موظف است پیغام خود را با همان قدرتی منتشر کند که پیش‌تر به سرخوشه CH_i پیغام *join* را ارسال کرده بود. در این مرحله، اگر نمونه آزمایشی پیغام خود را با قدرت متفاوتی ارسال کند، سرخوشه CH_i این موضوع را متوجه شده و آن را به عنوان گره بدخواه علامت می‌زند. البته، با توجه به اینکه مقدار *RSSI* ممکن است همیشه دقیق نباشد و به عوامل محیطی متعددی بستگی دارد، لذا می‌توان یک "**آستانه قابل تحمل**" در نظر گرفت و چنانچه اختلاف *RSSI* بسته‌های دریافت شده از یک گره عضو کمتر از این آستانه قابل تحمل بود، چشم‌پوشی شود.

هر سرخوشه‌ی CH_j پس از دریافت پیغام ES_{CH_i} ، ابتدا فاصله این گره (یعنی ES_{CH_i}) تا خودش را مطابق رابطه (۱) محاسبه نموده و در متغیری به نام *dis* قرار می‌دهد. سرخوشه CH_j سپس *member_table* خود را پیمایش نموده و هر عضو u که فاصله تا آن برابر *dis* باشد را انتخاب و به عنوان "**گره‌های بدخواه احتمالی**" برای سرخوشه CH_i ارسال می‌کند. اگر سرخوشه CH_j هیچ عضوی نداشته باشد که فاصله تا آن برابر *dis* باشد، در این صورت پاسخی برای سرخوشه CH_i نمی‌فرستد. سرخوشه CH_i چنانچه حداقل از T_{min} سرخوشه دیگر پاسخ دریافت کند، در این صورت لیست گره‌های بدخواه احتمالی دریافت کرده از دیگر سرخوشه‌ها به همراه نمونه آزمایشی خود، یعنی ES_{CH_i} را به عنوان "**گره‌های بدخواه سایبیل**" در لیستی در حافظه خود به نام *Sybil_List* ثبت می‌کند. در غیر این صورت هیچ گره بدخواهی در این مرحله شناسایی نمی‌شود. طبق مسئله "مکان‌یابی به کمک سه نقطه" [19]، آستانه T_{min} باید حداقل با مقدار 2 تنظیم شود تا دقت تشخیص بالا باشد. پس از پایان این مرحله (چه گره بدخواهی شناسایی شود یا نشود)، سرخوشه آغازگر CH_i این نمونه آزمایشی ES_{CH_i} را از لیست SL خود حذف کرده و یک عضو دیگر را از SL خود (در صورت وجود) به عنوان نمونه آزمایشی جدید (ES_{CH_i}) انتخاب نموده و با یک پیغام اطلاعات مربوط به نمونه آزمایشی جدید را برای دیگر سرخوشه‌ها ارسال می‌کند. دیگر سرخوشه‌ها با دریافت این پیغام، نمونه آزمایشی جدید سرخوشه CH_i را جایگزین نمونه آزمایشی قبلی آن در *ES_List* خود می‌کنند. به این ترتیب *ES_List* تمام سرخوشه‌ها با نمونه آزمایشی جدید برای سرخوشه CH_i بروزرسانی می‌شود. در این مرحله، اگر SL سرخوشه CH_i تهی باشد، مقدار 1- به عنوان نمونه آزمایشی برای دیگر سرخوشه‌ها ارسال می‌شود. مقدار 1- در *ES_List*ها به عنوان نمونه آزمایشی سرخوشه CH_i نشان دهنده این است که SL سرخوشه CH_i تهی شده و از این مرحله به بعد سرخوشه CH_i هیچ‌گاه به عنوان آغازگر روال تشخیص انتخاب نمی‌شود.

در مرحله بعد نیز، سرخوشه‌ای که دارای نمونه آزمایشی با بیشترین فاصله است (مطابق لیست *ES_List*ها) به عنوان سرخوشه آغازگر انتخاب می‌شود و این عملیات گفته شده در بالا دوباره تکرار می‌شود. روال تشخیص تا زمانی ادامه می‌یابد که تمامی عضوهای *ES_List* (در واقع *ES_List* تمام سرخوشه‌ها) 1- شود. در این زمان، اجرای الگوریتم پیشنهادی خاتمه یافته و هر سرخوشه لیست گره‌های موجود در *Sybil_List* خود را به عنوان گره‌های بدخواه به ایستگاه پایه و یا دیگر گره‌ها اعلام می‌کند. البته، در برخی حالات ممکن است برخی گره‌های قانونی به اشتباه توسط الگوریتم پیشنهادی به عنوان گره‌های سایبیل علامت زده شوند. به سناریوی زیر دقت کنید:

سناریو ۱: شبکه‌ای را در نظر بگیرید که به صورت شکل (۳) خوشه‌بندی شده است. در این‌جا، سه گره سرخوشه به نام‌های *H1*، *H2* و *H3* داریم. هم‌چنین یک گره بدخواه در شبکه وجود دارد که با شناسه *S1* به سرخوشه *H1*، با شناسه *S2* به سرخوشه *H2* و با شناسه *S3* به سرخوشه *H3* وصل شده است (پیغام

join ارسال کرده است). در مرحله خوشه‌بندی، سرخوشه *H1* فاصله تخمینی گره عضو *S1* تا خود را 11 تخمین می‌زند. هم‌چنین سرخوشه *H2* فاصله تا *S2* را 18 و *H3* فاصله تا *S3* را 5 تخمین می‌زند. در فاز اول (فاز پویش) الگوریتم پیشنهادی، سرخوشه‌ها *SL* خود را می‌سازند. به عنوان مثال، *SL* سرخوشه *H2* حاوی گره‌های $SL=\{S2,u,v\}$ خواهد بود چراکه فاصله تخمینی تا این گره‌ها بیشتر از $avg\_distance$=10.75 (میانگین فاصله سرخوشه *H2* تا عضوهایش) است. در فاز دوم، هر سرخوشه، گره عضوی که بیشترین فاصله تا آن دارد را به عنوان نمونه آزمایشی خود (*ES*) انتخاب می‌کند. سپس سرخوشه‌ها نمونه آزمایشی خود را برای یکدیگر ارسال می‌کنند. در این مثال، برای سرخوشه‌های *H1*، *H2* و *H3* به ترتیب گره‌های عضو *K* (یا *S1*)، *S2* و *F* (یا *G*) به عنوان نمونه آزمایشی انتخاب و در *ES_List*ها ثبت می‌شود. سپس، سرخوشه *H2* به عنوان آغازگر روال تشخیص انتخاب می‌شود چراکه دارای نمونه آزمایشی با بیشترین فاصله است. سرخوشه *H2* یک پیغام برای نمونه آزمایشی خود، یعنی *S2* ارسال می‌کند و به او دستور می‌دهد یک پیغام در شبکه منتشر کند. گره *S2* پیغام خود را با توانی ارسال می‌کند که گره‌هایی با فاصله حداکثر 18 واحد از آن، پیغام را دریافت کنند. از این‌رو، سرخوشه‌های *H1* و *H3* پیغام *S2* را دریافت و فاصله تخمینی خود تا این گره را محاسبه می‌کنند. سرخوشه *H1* فاصله تا *S2* را 11 و سرخوشه *H3* فاصله تا *S2* را 5 تخمین می‌زند. سرخوشه *H1* جدول *member_table* خود را پیمایش نموده و چون فاصله تا عضوهای *K* و *S1* را نیز برابر 11 می‌بیند، لذا این دو عضو خود را به عنوان "گره‌های بدخواه احتمالی" به سرخوشه آغازگر، یعنی *H2* ارسال می‌کند. به همین ترتیب سرخوشه *H3* نیز گره‌های عضو *W* و *S3* را به عنوان "گره‌های بدخواه احتمالی" به سرخوشه *H2* ارسال می‌کند. با فرض این‌که آستانه T_{min}=2 باشد، در این صورت سرخوشه *H2* گره‌های *S2* (نمونه آزمایشی خود)، *K*، *S1*، *W* و *S3* را به عنوان گره‌های بدخواه سایبیل در *Sybil_List* خود ثبت می‌کند. در نتیجه گره‌های *W* و *K* به اشتباه به عنوان گره‌های بدخواه سایبیل تشخیص داده شده‌اند. باید توجه شود در این سناریو چنان‌چه T_{min}>2 انتخاب شده باشد، گره‌های بدخواه شناسایی نمی‌شوند. پس به‌طور کلی، T_{min} باید کوچکتر از تعداد کلاسترها در شبکه (یعنی C_{opt}) باشد.



شکل ۳: یک حالت ممکن از خوشه‌بندی شبکه و تشخیص غلط الگوریتم پیشنهادی

۵- ارزیابی کارایی و نتایج شبیه‌سازی

۵-۱- سربار ارتباطات

هر سرخوشه، لیست *SL* خود را ساخته و نمونه آزمایشی، *ES*، را بدست می‌آورد. سپس، همه سرخوشه‌ها *ES* خود را برای یکدیگر منتشر (*broadcast*) می‌کنند (با سربار ارتباطاتی C_{opt}). سپس، هر سرخوشه با توجه به نمونه‌های آزمایشی خود و دیگر سرخوشه‌ها اقدام به ساخت *ES_List* می‌کند. از این مرحله به بعد، برای هر نمونه آزمایش، عملیات زیر انجام می‌گیرد:

سرخوشه آغازگر یک پیغام برای نمونه آزمایشی خود ارسال می‌کند و نمونه آزمایشی نیز یک بسته در شبکه منتشر (*broadcast*) می‌کند که این منجر به ارسال 2 بسته در شبکه می‌شود. بنابر این، به ازای هر نمونه آزمایشی، 2 بسته در شبکه منتشر می‌شود. تعداد کل نمونه‌های آزمایشی نیز برابر است با:

$$|ES| = C_{opt} \times \frac{N+M(S-1)}{2\times C_{opt}} = \frac{N+M(S-1)}{2} \qquad (۲)$$

با فرض این‌که بسته ارسال شده از سوی نمونه آزمایشی، در حالت میانگین، به دست $\frac{C_{opt}-1}{2}$ سرخوشه دیگر (تحت عنوان **سرخوشه‌های گیرنده**) برسد، هر یک از این سرخوشه‌های گیرنده، جدول *member_table* خود را پویش نموده و در صورت لزوم یک پیغام حاوی لیست "گره‌های بدخواه احتمالی" برای سرخوشه آغازگر ارسال می‌کند.

احتمال سایبیل بودن یک گره در شبکه $\frac{M\times S}{N+M(S-1)}$ است. از طرفی هم، یک گره سایبیل عضو یک سرخوشه، نظیر CH_i، تنها زمانی‌که در فاصله دورتر از "میانگین فاصله سرخوشه تا عضوها" واقع شده باشد، در لیست گره‌های مشکوک (*SL*) سرخوشه CH_i قرار می‌گیرد. بنابراین، با توجه به فرض هوشمند عمل کردن گره بدخواه در انتخاب سرخوشه‌ها، این احتمال وجود دارد که یک گره سایبیل عضو یک سرخوشه، در لیست *SL* آن قرار نگیرد. فرض می‌کنیم، به‌طور میانگین، نیمی از گره‌های سایبیل عضو یک سرخوشه، در لیست *SL* آن سرخوشه قرار خواهند گرفت. بنابر این، احتمال بدخواه بودن نمونه آزمایشی انتخاب شده توسط سرخوشه آغازگر، E_m، برابر است با:

$$E_m = \frac{\frac{M\times S}{2}}{N+M(S-1)} \times \frac{N+M(S-1)}{2\times C_{opt}} = \frac{M\times S}{(N+M(S-1))(2\times C_{out})} \qquad (۳)$$

حال، احتمال این‌که گره بدخواه (همان نمونه آزمایشی انتخاب شده توسط سرخوشه آغازگر) همزمان عضو هر یک از سرخوشه‌های دریافت کننده نیز باشد، E_C، برابر است با:

$$E_C = \frac{S-1}{C_{opt}-1} \qquad (۴)$$

در این صورت، به تعداد $\frac{C_{opt}-1}{2}\times E_C \times 1$ بسته در این مرحله توسط سرخوشه‌های دریافت کننده به سوی سرخوشه آغازگر ارسال می‌شود. در مرحله بعد، سرخوشه آغازگر، نمونه آزمایشی انتخاب شده را از *SL* خود حذف نموده و نمونه آزمایشی جدید را درصورت وجود انتخاب و با یک پیغام آن را برای دیگر سرخوشه‌ها ارسال می‌کند (با سربار ارتباطی ۱ بسته). مراحل ذکر شده در بالا به ازای کل نمونه‌های آزمایش، یعنی $|ES|$ مرحله تکرار می‌شود. از این‌رو، کل سربار ارتباطات به ازای اجرای الگوریتم پیشنهادی برابر است با:

$$Comm_{cost} = C_{opt} + |ES|\times 2 + |ES|\times(E_m \times E_C \times \frac{C_{opt-1}}{2}\times 1) + |ES|\times 1$$

$$\Rightarrow Comm_{cost} = C_{opt} + \frac{N+M(S-1)}{2}\times \qquad (5)$$

$$\left[2+1+(\frac{M\times S}{(N+M(S-1))(2\times C_{opt})}\times\frac{S-1}{2})\right]$$

۵-۲- نتایج شبیه‌سازی

الگوریتم پیشنهادی پیاده‌سازی گردیده و با انجام تعدادی آزمایش، کارایی آن با دیگر الگوریتم‌ها مقایسه شده است. معیارهای مورد ارزیابی عبارتند از: **نرخ تشخیص درست:** درصدی از گره‌های سایبیل است که توسط یک الگوریتم امنیتی شناسایی می‌شود، **نرخ تشخیص غلط:** درصدی از گره‌های نرمال که به اشتباه توسط الگوریتم امنیتی به‌عنوان گره‌های سایبیل شناسایی می‌شود، **سربار ارتباطات:** تعداد بسته‌های ارسالی توسط کل گره‌ها، به ازای اجرای یک الگوریتم امنیتی در شبکه است .

در اجرای شبیه‌سازی‌ها، فرض می‌شود شبکه حاوی N گره حسگر است که از این تعداد، M گره بدخواه می‌باشند. گره‌ها به‌طور تصادفی در یک ناحیه ۱۰۰×۱۰۰ مترمربع پراکنده شده‌اند. شبکه به C_{opt} خوشه تقسیم می‌شود. هر گره بدخواه، S شناسه جعلی از خود منتشر می‌کند ($S \le C_{opt}$). به منظور اطمینان از اعتبار نتایج، هر شبیه‌سازی ۱۰۰۰ بار تکرار شده و نتیجه نهایی از میانگین نتایج این ۱۰۰۰ تکرار بدست آمده است.

آزمایش ۱: در این آزمایش، پارامترهای N=100، M=1، $C_{opt}=7$ تنظیم شده و S از 2 تا 7 تغییر داده شده و نتایج حاصل در قالب معیارهای نرخ تشخیص درست، نرخ تشخیص غلط و سربار ارتباطات به ترتیب در شکل‌های (۴)، (۵) و جدول (۱) آمده است. همان‌طور که از نتایج این آزمایش در شکل (۴) مشخص است، هرچه گره بدخواه همزمان عضو خوشه‌های بیشتری شود، نرخ تشخیص درست الگوریتم پیشنهادی افزایش می‌یابد. به عنوان مثال، زمانی‌که گره بدخواه عضو دو خوشه باشد، نرخ تشخیص ۸۶% و زمانی که عضو بیش از ۳ خوشه باشد، نرخ تشخیص بالاتر از ۹۹% می‌شود. دو دلیل برای این موضوع وجود دارد. دلیل اول، از آنجا که در الگوریتم پیشنهادی سرخوشه‌ها با همکاری همدیگر اقدام به شناسایی گره بدخواه می‌کنند لذا هرچه گره بدخواه همزمان عضو خوشه‌های بیشتری باشد، نرخ تشخیص افزایش می‌یابد. دلیل دوم، این است که اگر گره بدخواه همزمان عضو تعداد خوشه‌های کمتری باشد، احتمال شناسایی شدن آن توسط الگوریتم پیشنهادی کاهش می‌یابد. چراکه گره بدخواه می‌تواند خوشه‌هایی را جهت عضویت انتخاب کند که در نزدیک‌ترین فاصله با آن هستند که این سبب می‌شود گره بدخواه در لیست SL هیچ یک از سرخوشه‌ها قرار نگیرد. ولی اگر گره بدخواه همزمان عضو خوشه‌های بیشتری شود، احتمال این‌که فاصله‌اش با تمام این سرخوشه به قدری کم باشد که در لیست SL هیچ یک از این سرخوشه‌ها قرار نگیرد کاهش می‌یابد. از این‌رو احتمال تشخیص آن افزایش می‌یابد.

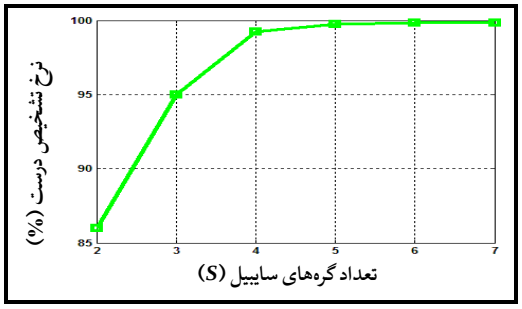
نتایج این آزمایش در شکل (۵) نشان می‌دهد نرخ تشخیص غلط الگوریتم پیشنهادی به ازای S=2 و S=3 به ترتیب ۰٫۰۶% و ۰٫۰۷% و به ازای $S>3$ نرخ این معیار تقریبا ۰٫۰۸۵% است. همان‌طور که در سناریو ۱ شرح داده شد، در هر دور اجرای الگوریتم تشخیص برای یک نمونه آزمایشی، ممکن است بعضی گره‌های قانونی به اشتباه به عنوان گره‌های سایبیل شناسایی شوند. لذا، هرچه گره بدخواه عضو خوشه‌های بیشتری باشد، تعداد نمونه‌های آزمایشی افزایش یافته و در نتیجه نرخ تشخیص غلط نیز تا حد اندکی افزایش می‌یابد.

هم‌چنین، در جدول (۱)، هر دو نتایج محاسباتی و شبیه‌سازی سربار ارتباطات ارائه شده است. نتایج محاسباتی نشان داد با افزایش S (یک واحد)، سربار ارتباطات ۱ یا ۲ واحد افزایش می‌یابد و نتایج شبیه‌سازی نشان داد با افزایش S، سربار ارتباطات ۲ یا ۳ واحد افزایش می‌یابد. بنابر این، تغییر پارامتر S تأثیر چندانی بر میزان سربار ارتباطات الگوریتم پیشنهادی ندارد.

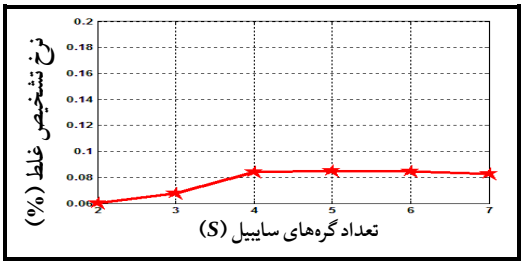
آزمایش ۲: در این آزمایش، پارامترهای S=5، M=1، $C_{opt}=7$، T_{min}=2 تنظیم شده و N از 100 تا 500 (با گام افزایش 100) تغییر داده شده و تأثیر آن بر کارایی الگوریتم پیشنهادی ارزیابی شده است. شکل (۶) نرخ تشخیص درست، شکل (۷) نرخ تشخیص غلط و جدول (۲) سربار ارتباطی حاصل از این آزمایش را نشان می‌دهد. شکل‌های (۶) و (۷) نشان می‌دهد افزایش تعداد گره‌ها در شبکه منجر به افزایش نرخ تشخیص درست و غلط الگوریتم پیشنهادی می‌شود. چراکه افزایش تعداد گره‌ها در شبکه منجر به افزایش تعداد نمونه‌های آزمایشی و در نتیجه افزایش تعداد دورهای اجرای الگوریتم تشخیص می‌شود. نتیجه این آزمایش نشان داد به ازای مقادیر مختلف N، نرخ تشخیص درست بیشتر از ۹۹٫۷%و نرخ تشخص غلط کوچکتر از ۰٫۰۸۶% است.

نتیجه این آزمایش در جدول (۲) نشان می‌دهد، با افزایش تعداد گره‌ها در شبکه، سربار ارتباطات نیز افزایش می‌یابد. چراکه افزایش تعداد گره‌ها منجر به افزایش تعداد نمونه‌های آزمایشی و در نتیجه افزایش تعداد دورهای اجرای الگوریتم تشخیص می‌شود. از این‌رو، سربار ارتباطات نیز افزایش خواهد یافت. نتایج محاسباتی نشان می‌دهد به ازای افزایش N (100 واحد)، سربار ارتباطات 150 واحد افزایش می‌یابد و نتایج شبیه‌سازی نشان داد به ازای این میزان افزایش در پارامتر N، سربار ارتباطات تقریبا 143 واحد افزایش می‌یابد.

آزمایش ۳: از آنجا که مدل حمله درنظر گرفته شده در اینجا متفاوت از مدل حمله درنظر گرفته شده در دیگر الگوریتم‌هاست لذا نمی‌توان کارایی الگوریتم پیشنهادی را با دیگر الگوریتم‌های موجود در قالب معیارهای نرخ تشخیص و نرخ تشخیص غلط (به ازای تغییر در پارامترهای مختلف نظیر M، S، C_{opt}) مقایسه نمود. ولی در حالت میانگین می‌توان این مقایسه را انجام داد. از این‌رو، در این آزمایش، میانگین نرخ تشخیص درست و میانگین نرخ تشخیص غلط الگوریتم پیشنهادی و دیگر الگوریتم‌های موجود مقایسه گردیده است. جدول (۳) نتیجه این مقایسه را نشان می‌دهد. همان‌طور که از نتیجه این مقایسه مشخص است، الگوریتم پیشنهادی با میانگین نرخ تشخیص درست ۹۹٫۸% پایین‌تر از الگوریتم‌های مطرح شده در [11] و [17] و برتر از سایر الگوریتم‌هاست. البته الگوریتم مطرح شده در [11] به دلیل متکی بودن به ارسال و حل پازل‌های مشتری، سربار ارتباطی و محاسباتی بیشتری نسبت به الگوریتم پیشنهادی به شبکه تحمیل می‌کند. اما از حیث میانگین نرخ تشخیص غلط، الگوریتم پیشنهادی با نرخ ۰٫۱% برتر از سایر الگوریتم‌ها می‌باشد.



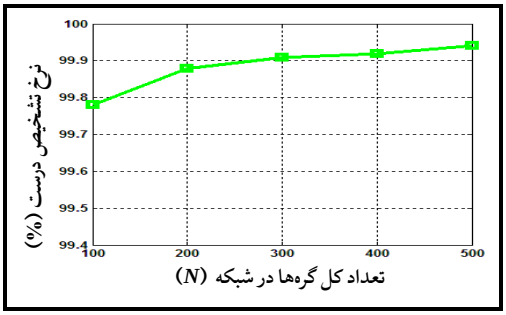
شکل ٤: تأثیر پارامتر S بر نرخ تشخیص درست الگوریتم پیشنهادی



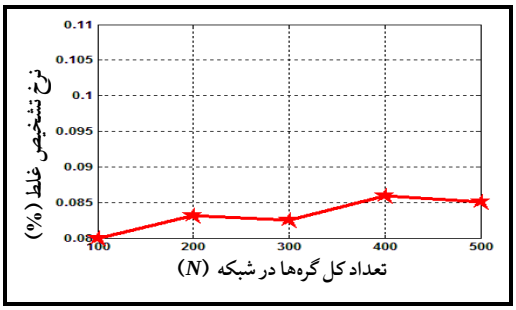
شکل ۵: تأثیر پارامتر S بر نرخ تشخیص درست الگوریتم پیشنهادی

جدول ۱: تأثیر پارامتر S بر سربار ارتباطات الگوریتم پیشنهادی

	S=2	S=3	S=4	S=5	S=6	S=7
نتایج محاسباتی	158	160	161	163	165	167
نتایج شبیه‌سازی	143	145	147	149	152	155



شکل ٦: تأثیر پارامتر N بر نرخ تشخیص درست الگوریتم پیشنهادی



شکل ۷: تأثیر پارامتر N بر نرخ تشخیص غلط الگوریتم پیشنهادی

جدول ۲: تأثیر پارامتر N بر سربار ارتباطات الگوریتم پیشنهادی

	N=100	N=200	N=300	N=400	N=500
نتایج محاسباتی	163	313	463	613	763
نتایج شبیه‌سازی	149	291	434	577	721

جدول ۳: مقایسه کارایی الگوریتم پیشنهادی با دیگر الگوریتم‌ها در قالب معیارهای میانگین نرخ تشخیص درست/غلط

الگوریتم	میانگین نرخ تشخیص درست	میانگین نرخ تشخیص غلط
[6]	92%	2%
[7]	90%	1%
[11]	100%	6%
[12]	99%	5%
[13]	98%	6%
[17]	100%	5%
[18]	99%	5%
الگوریتم پیشنهادی	99.8%	0.08%

۶- نتیجه‌گیری

در این مقاله، یک مدل جدید از برپایی حمله *Sybil* در شبکه‌های حسگر بی‌سیم مبتنی بر خوشه‌بندی، نظیر *LEACH*، پیشنهاد گردید. سپس یک الگوریتم، مبتنی بر مشخصه قدرت سیگنال دریافتی و مکان‌یابی به کمک سه نقطه جهت مقابله با این مدل جدید از حمله *Sybil* مطرح شد. الگوریتم پیشنهادی شبیه‌سازی گردید و نتیجه آزمایش‌ها نشان داد الگوریتم پیشنهادی با تحمیل سربار ارتباطی اندکی به شبکه، قادر به شناسایی ۹۹٫۸%گره‌های *Sybil* و نرخ تشخیص غلط ۰٫۰۸% است (در حالت میانگین).

منابع

[1] Yick J., Mukherjee B. and Ghosal D., "Wireless sensor network survey", in: Proc. of the Computer Networks 52, pp. 2292–2330, 2008.

[2] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energyefficient communication protocol for wireless microsensor networks. In IEEE Hawaii Int. Conf. on System Sciences, pages 4–7, 2000.

[3] Douceur J. R., "The Sybil attack", in: Proc. of the First International Workshop on Peer-to-Peer Systems (IPTPS '02), 2002.

[4] Karlof C. And Wagner D, "**Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures**", in: Proc. of the AdHoc Networks, pp. 299-302, 2003.

[5] Newsome J. and et al., "**The Sybil attack in sensor networks: analysis and defenses**", in: Proc. of the International Symposium on Information Processing in Sensor Networks, pp. 259–268, 2004.

[6] Chen S. and et al., "**A Security Routing Mechanism against Sybil Attack for Wireless Sensor Networks**", in: Proc. of the International Conference on Communications and Mobile Computing, 2010.

[7] Jangra A., Swati, Priyanka, "**Securing LEACH Protocol from Sybil Attack using Jakes Channel Scheme (JCS)**", in: Proc. of the International Conferences on Advances in ICT for Emerging Regions(ICTer2011), 2011.

[8] Vasudeva1 A. and Sood M., "**SYBIL ATTACK ON LOWEST ID CLUSTERING ALGORITHM IN THE MOBILE AD HOC NETWORK**", in: Proc. of the International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.5, 2012.

[9] J. Zhao and R. Govindan, (2003), "Understanding Packet Delivery Performance in Dense Wireless Sensor Networks", In Proc. ACM Sensys.

[10] Zhong S. and et al., "**Privacy-preserving location based services for mobile users in Wireless Networks**", In: Proc. of the Technical Report YALEU/DCS/TR-1297, Yale Computer Science, 2004.

[11] Demirbas M. and Song Y., "**An RSSI-based scheme for Sybil attack detection in wireless sensor networks**", In: Proc. of the IEEE Computer Society International Symposium on World of Wireless, Mobile and Multimedia Networks, pp. 570–574, 2006.

[12] Ssu K. F, Wang W. T. and Chang W. C., "**Detecting Sybil attacks in wireless Sensor Networks using neighboring information"**, in: Proc. of the Computer Networks 53, pp. 3042–3056, 2009.

[13] Misra S. and Myneni S., "**On Identifying Power Control Performing Sybil Nodes in Wireless ensor Networks Using RSSI**", in: Proc. of the IEEE Global Telecommunications Conference (GLOBECOM 2010), 2010.

[14] ZHANG Y., FAN K.-F., ZHANG S.-B. and MO W., "**AOA based trust evaluation sheme for Sybil attack detection in WSN**", in: Proc. of the journal on Application Research of Computers, 2010.

[15] Li F., Mittal P., Caesar M. and Borisov N., "**SybilControl: Practical Sybil Defense with Computational Puzzles**", in: Proc. of the Networking and Internet Architecture, Jan 2012.

[16] WANG X.-D., SUN Y.-Q. and MENG X.-X., "**Cluster-based Defending Mechanism for Sybil Attacks in Wireless Sensor Network**", in: Proc. of the Computer Engineering;2009 .

[17] Jamshidi, M., Esnaashari, M. and Meybodi, M. R., "**An Algorithm for Defending Sybil Attacks based on Client Puzzles and Learning Automata for Wireless Sensor Networks**", in: Proceeding of 18th National Conference of Computer Society of Iran , Sharif University, Tehran, Iran, March 14-16, 2013.

[18] Jamshidi, M., Esnaashari, Nasri A., Hanani A. and Meybodi, M. R., "**Detecting Sybil Nodes in Mobile Wireless Sensor Networks using Observer Nodes**", in: Proceeding of 10th International ISC Conference On Information Security & Cryptologhy, Computer Society of Iran , yazd University, yazd, Iran, August 29-30, 2013.

[19] **http://www.kowoma.de/en/gps/positioning.htm**

پانویس‌ها

[1] *Received Signal Strength Indicator*

[2] *Suspicious List*

[3] *Experimental Sample*