

학번 20046  
이름: 박건호

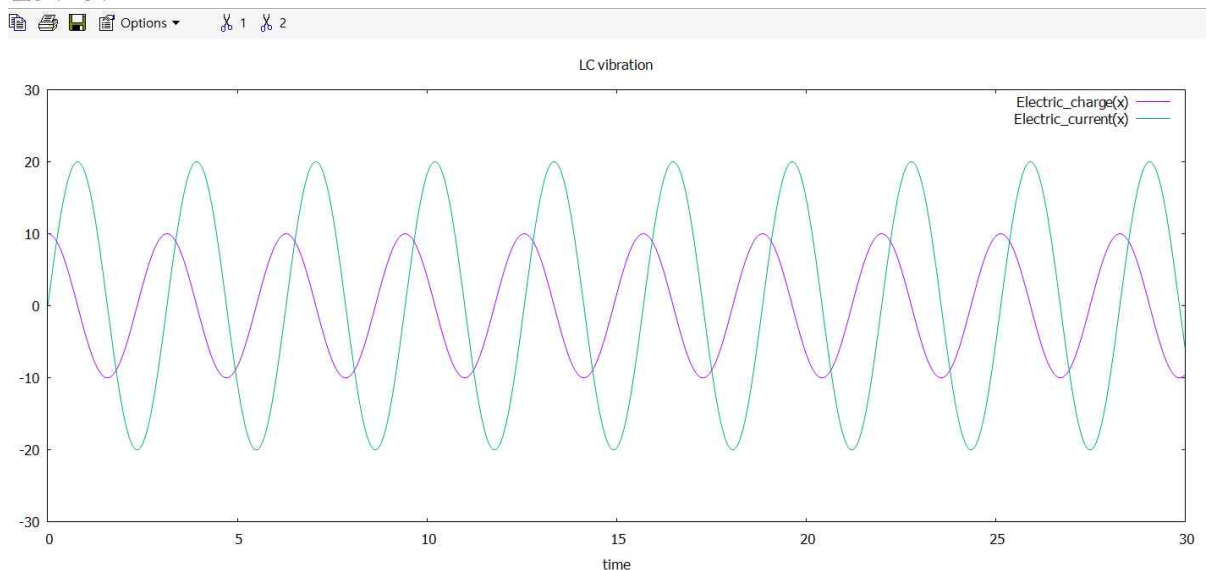
1-1 2차원 (x축, y축) 그래프 만들어보기

제목: LC 진동

관련 수학/ 과학 교과명 : 물리

관련 내용 또는 주제 : 인덕터와 축전기 사이의 진동(LC 진동)


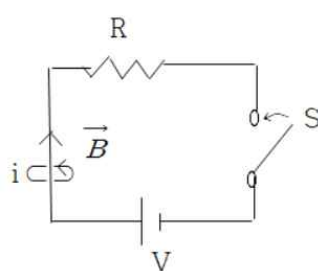

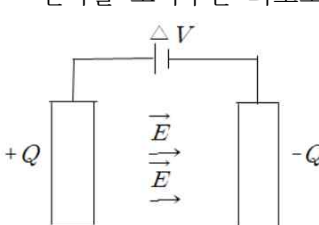
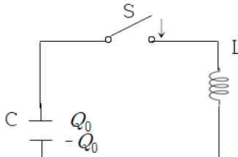
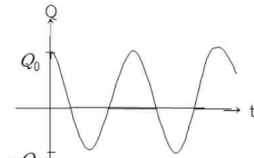
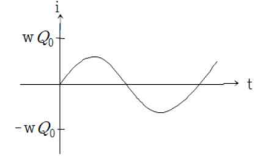
만든 이차원 그래프 결과 캡처 사진



만든 것의 과학/수학/정보 관점의 의미

LC 진동은 축전기 와 인덕터 사이의 진동을 의미하며 이는 도선의 저항이 0이라는 가정하에 계속 전류와 전하량이 sin,cos그래프를 그리며 반복되는 형태가 나온다. 이는 축전기와 인덕터가 가지는 에너지를 계산하여 더하면 일정한 결과가 나와 무한 반복되는 형태가 나오며 이는 무선 통신과 라디오에서 많이 활용된다.

# LC진동 ( LC Vibration )이란? + 미분방정식 푸는 방법

<p>⇒ L : (인덕터)</p>  <p>인덕터란? (=유도기) ≈ 코일이다. 자체유도현상을 극대화한 회로소자이다.</p>  <p>i 생성 → B 생성 ⇒ 자속생성 (전자기유도에 의한 자체유도기전력 <math>V_L</math> 형성)</p> <p>여기서 <math>V_L = -L \frac{di}{dt}</math>라고 정의한다. ( L : 자체인덕턴스, [H] )</p>	<p>⇒ C : (축전기)</p>  <p>축전기란? (배터리) 전하를 모아주는 회로소자이다.</p>  <p><math>C = \frac{\Delta Q}{\Delta V}</math> [F]</p> <p>⇒ <math>V = \frac{Q}{C}</math> 이다.</p>
<p>* 키리히호프 법칙</p> <p>① 분기점법칙 ( <math>\sum i_{in} = \sum i_{out}</math> ) ② 폐회로법칙 ( <math>\sum V_i = 0</math> )</p>	<p>* LC회로란?</p> <p>인덕터(L)와 축전기(C)가 같이 있는 회로를 의미한다.</p>
 <p><math>i = - \frac{dq}{dt}</math> ..... ①</p>	<p>키리히호프 법칙에 의해</p> <p><math>\frac{q}{c} - L \frac{d_i}{d_t} = 0</math> 이다.</p> <p>①번 공식을 대입시키면</p> <p><math>\frac{dq^2}{d^2t} + \frac{1}{LC}q=0</math> (<math>W=\frac{1}{\sqrt{LC}}</math>) : 고유각진동수</p> <p>∴ <math>q(t) = Q_0 \cos wt</math> <math>i(t) = w Q_0 \sin wt</math></p>
	 <p><math>T = 2\pi \sqrt{LC}</math> 이다.</p>
<p>여기서 <math>U_c = \frac{Q^2}{2C} = \frac{Q_0^2}{2C} \cos^2 wt</math></p> <p>∴ <math>U_c + U_L = \frac{Q^2}{2C}</math> (일정)</p>	<p><math>U_L = \frac{1}{Q} Li^2 = \frac{1}{2} Lw^2 Q_0^2 \sin^2 wt = \frac{Q_0^2}{2C} \sin^2 wt</math></p> <p>⇒ 계속 반복한다.</p>

- 자신이 분석해 변경하거나 바꾼 것들

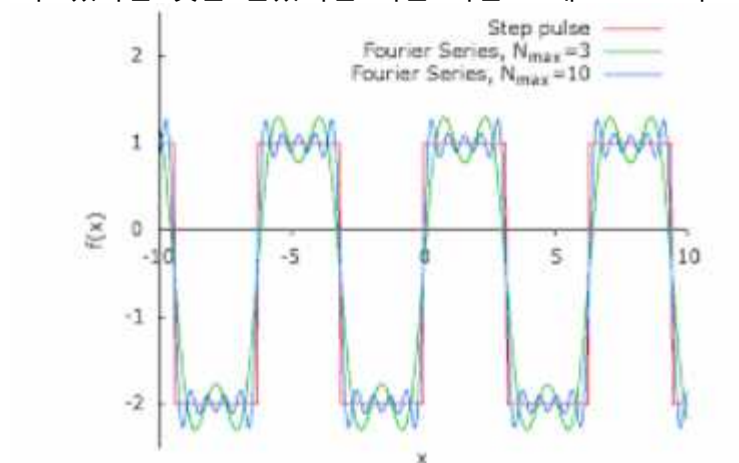
제가 직접 생각하며 유튜브 , 블로그에 있는 내용들을 보고 처음부터 만들었습니다.

- 참고한 예시 또는 내용

참고한 내용은 블로그인데 기본적인 명령어인 라벨 붙이는 방법, 제목 붙이는 방법, xy범위 설정하는 방법, 여러 그래프를 한번에 표현하게 해주는 multiplot의 소개들이 있는 블로그를 보았다.처음에 무엇을 만들지 고민하다가 전기 회로에 관하여 결과만 배웠는데 시간이 지남에 따라서 정확히 전하량이나 전류값이 어떻게 바뀌는지 그래프로 나타내면 좋을 것 같아 만들어 보았습니다. 그리고 LC 진동을 구현하는 과정에서 이게 미분 방정식을 푸는 방법에 대한 방법을 찾아보았다.

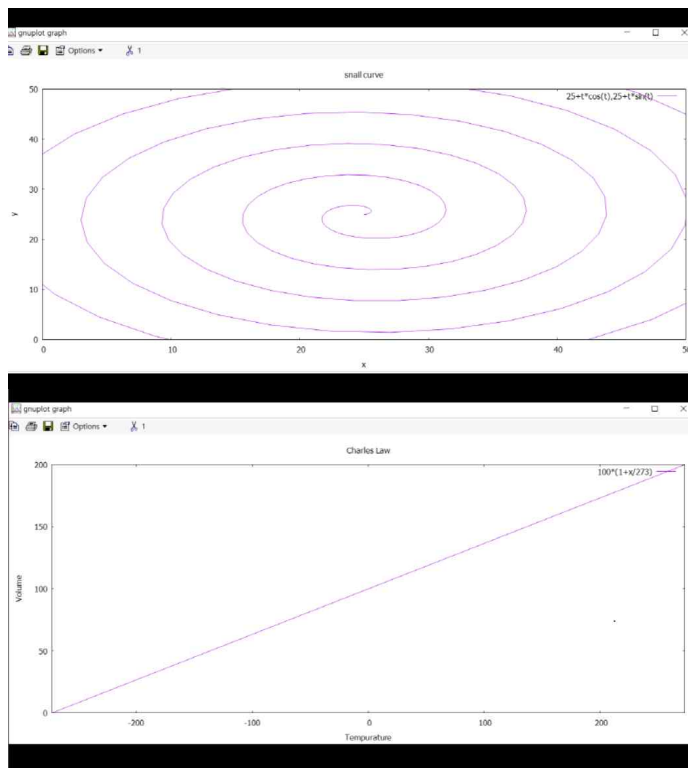
찾아보면서 재미있었던 예시와 그 이유는?

gnuplot에서 선만 표현한 것이 아닌 어떤 면을 색깔을 이용해 표현한 것이 신기하였는데 2차원그래프를 그리는데 선이 아닌 면의 요소를 생각해 낸 것이 신기하기도 했고 색깔이 하나가 아닌 부드럽게 색깔이 바뀌도록 구현을 할 수 있다는 것이 놀라웠기도 하였다. 그리고 찾아본 예시 중에 푸리에 급수를 gnuplot을 이용하여 나타낸 것을 보았는데 연속함수를 많이 합쳐 불연속 함수처럼 보이도록 근사시킬 수 있다는 것은 알았지만 이를 직접 그래프로 보니 되게 신기하였다.



- 만들면서 생각해보거나 시도해 본 것들

처음에는 만들것이 많이 떠오르지 않아서 샤를 법칙, 보일 법칙, 매개변수를 이용한 달팽이 함수등과 같이 간단한 함수들을 만들어보았다. 하지만 너무 직선 반비례 그래프 1개여서 너무 간단한 생각이 들어 매개변수 들을 이용하여서 파동을 만들어 보았다.



- 만들면서 이해하게 되었거나 느낀 점들

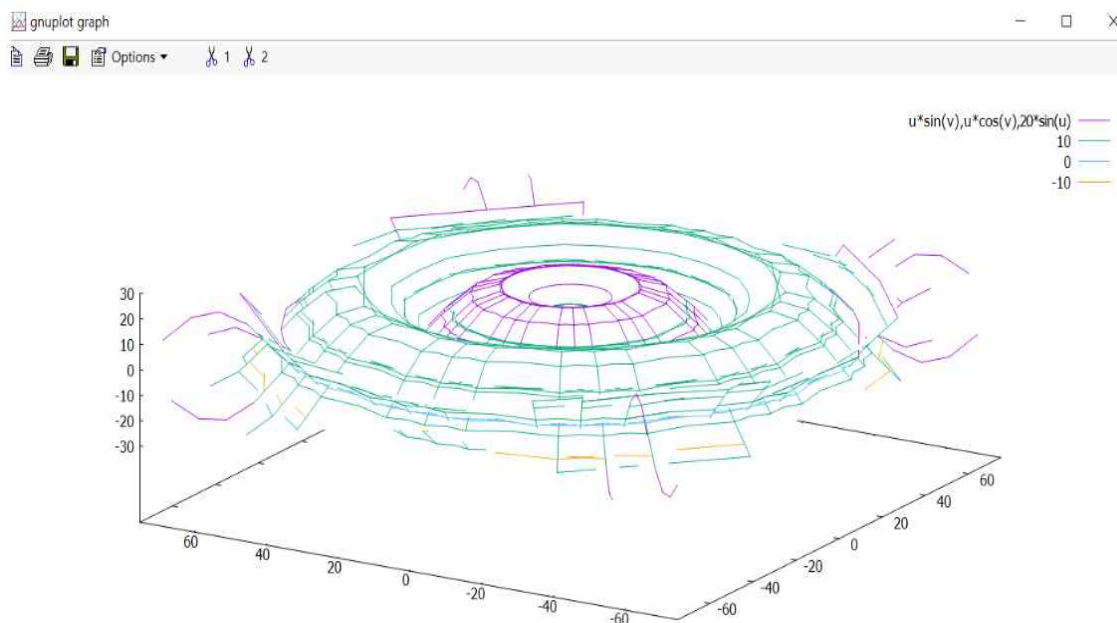
처음에 제가 진로가 수학 쪽이고 연구 활동에 관련된 것으로 만들려고 하였으나 연구 주제가 Bexier curve에 관한 것이었고 gnuplot 홈페이지에 그것에 관한 내용이 직접적으로 나와 있어서 이를 대상으로 만들지 못한 것이 아쉬웠다. 하지만 연구를 진행하면서 새로운 이론들이 나와서 확장 시킨다면 이를 gnuplot으로 만들어 볼것이라고 생각하였다.

## 1-2. 3차원(x축, y축, z축) 그래프 만들어보기

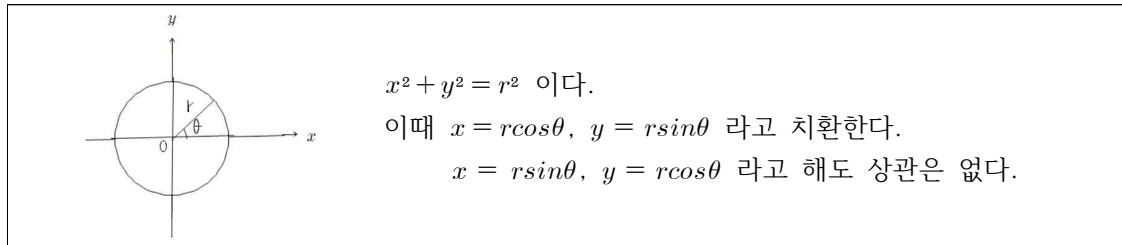
### 만든 것에 대한 설명(결과 그림 캡처 후 삽입)

제목: 물결파

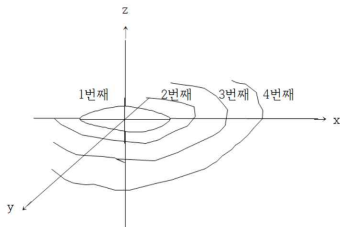
- 관련 수학/과학 교과명 : 물리
- 관련 내용 또는 주제 : 잔잔한 물에 물방울을 떨어뜨렸을 때 표면의 움직임
- 만든 3차원 그래프 결과 캡처 사진



극좌표



그런데 여기서 3차원으로 확장시켜서 표현하려면.



⇒ 각 위치에 따라 Z값을 다르게 해줘야 한다.

⇒ 각 번호에 따라 Z값도 물결을 만들도록  $\sin(\phi)$ 를 이용하자.

↳ 진폭은 20으로 정하여  $20 \times \sin(\phi)$  로 만들자.

∴ (  $u \times \sin(v)$ ,  $u \times \cos(v)$ ,  $20 \times \sin(u)$  )로 만들면 다음과 같은 그래프가 나온다

만든 것의 과학/수학/정보 관점의 의미

물결파는 파동의 종류 중에서 바다, 강과 같이 물이 많은 곳에서 나타나는 파동이다. 파동은 주로  $\sin, \cos$  그래프가 그려지며 물결파에서는 모든 방향으로 진행할 수 있는 특징이 있어 중심으로부터 원 모양으로 퍼져나간다. 만약에 중간에 어떤 물체를 넣으면 회절이 일어나는데 이는 표현하기 너무 어려워서 하지 않았다.

- 자신이 분석해 변경하거나 바꾼 것들

처음에는  $x, y, z$ 좌표로 이를 만들려고 하였으나 만들기 너무 힘들어 매개변수를 도입하였다.  $x^2 + y^2 = r^2$ 인 것을 활용하여 원의 모양을 만들었으며 거기에 하나의 축을 추가하여 위로 올라갔다 내려갔다 하는 것을 구현했다. 즉, 카테시안 좌표계로 이를 표현하려고 하였으나 이에 실패하여 극좌표를 이용하여 표현하였다.

- 참고한 예시 또는 내용

물결파와 관련된 구글 이미지를 보고 기초적인 지식은 블로그를 통해 알고 이 코드를 짰습니다. 다른 사람들은 어떤 과학적인 식에 대한 2변수함수에 대한 값을 표현하는 것을 많이 하여서 저는 이와 다르게 어떤 실생활에서 보이는 모양이나 형태를 만들어보고 싶었는데 물결파가 떠올라서 이것을 만들었습니다.

- 찾아보면서 재미있었던 예시와 그 이유는?

확실히 3차원으로 가니 면을 채우거나 그리는 것이 복잡한 것들이 많아서 재미있었던 것이 많았는데 별이나 하트모양을 극좌표를 이용해 구현한 것이 신기하였다. 극좌표라는 것이 익숙하지 않아 하나의  $x$ 좌표에 대해 여러  $y$ 좌표가 찍히고 곡선 모양도 있고 면의 모양도 있는데 그것을 어떻게 하나의 식으로 다 표현하였지 라는 생각도 있어 신기했다.

- 만들면서 생각해보거나 시도해 본 것들

만들면서 그래프가 이상하게 나타나는 것을 보고 어떻게 된건지 많이 고민하였는데 `isosamples`를 이용해 sample수를 늘렸더니 전체적인 모양이 나타나는 것을 보고 신기하였다. 그리고 면에다가 등고선 같은 것을 그리고 싶어서 고민했는데 `contour surface`라는 것이 실제로 있어서 적용시켜보기도 하였다.



- 만들면서 이해하게 되었거나 느낀 점들

gnuplot이 확실하게 java와 c언어와 같은 느낌이 아니어서 너무 코드 짜는 방식이나 내용이 달라 이해하기 어려웠던 것 같다. 그리고 3차원 그래프에서 매개변수를 이용하여 그래프를 더 쉽게 표현하였는데 편해져서 기분이 좋았다. 그리고 이렇게 있으면 좋겠다라고 생각한 것이 명령어로 모두 있는 것이 신기하였고 이를 하나씩 찾아가면서 하는 과정이 힘들었지만 다하고 나니 c언어보다 확실히 뿌듯하였다.