

제 3 장 시스템 트레이딩 따라하기

투자의 핵심 원칙:

고수익은 고위험을 감수해야 얻을 수 있다.

(High Risk, High Returns)

이 장에서는 제 2장에서 설명한 전략 개발과정에 따라 매매 전략을 개발하기 위한 코딩 및 평가 등을 해보는 부분이다. 초보자라면 이 장을 통해서 매매 전략이 어떻게 만들어지는가를 이해하게 될 것이다.

시스템 트레이딩의 가장 관심이 되는 선물을 이용한 데이트레이딩 전략과 단기 매매 전략에 중점을 두어 설명하기로 하겠다.

우선 시스템 트레이딩 매매 전략을 개발하는 과정에서 몇 가지 알아두어야 할 사항이 있다.

진입이나 청산, 자산 관리에서 공통적으로 많이 사용되는 함수식, 기본언어 등이 그것이다. 개발 과정에 들어 가기 전에 이 부분부터 알아보기로 하자.

1. 꼭 알아 두어야 할 것들

시스템 트레이딩 전략은 가격 및 거래량을 이용한 매매 전략들이 많은 편이다. 따라서 분간 차트를 이용하든지 일간 차트를 이용하든지, 가격 및 거래량을 표현하는 방법을 아는 것이 중요하다.

〈표 0-1〉 가격 표시 방법

구분	해당 봉	일간 기준	주간 기준	월간 기준	년간 기준
시가	Open	OpenD	OpenW	OpenM	OpenY
고가	High	HighD	HighW	HighM	HighY
저가	Low	LowD	LowW	LowM	LowY
종가	Close	CloseD	CloseW	CloseM	CloseY
거래량	Volume	VolumeD	VolumeW	VolumeM	VolumeY
비고		분봉, 일봉에서 사용	분봉, 일봉, 주 봉에서 사용	분봉, 일봉, 주 봉, 월봉에서 사용	분봉, 일봉, 주 봉, 월봉, 년봉 에서 사용
<p>예 1) 분봉에서 전일의 일봉 종가가 일봉 시가보다 크고, 오늘의 시가 보다 현재 봉의 종가가 크면.....</p> <p style="padding-left: 40px;">if closed(1) > opend(1) and close > opend then</p> <p>예 2) 일봉에서 전주의 종가가 시가보다 크고, 금주의 현재 고가가 전주의 고가보다 크면.....</p> <p style="padding-left: 40px;">if closew(1) > openw(1) and high > highw(1) then</p>					

위와 같은 규약에 따라서 각각의 시간대뿐만 아니라, 각 주기별 주가 및 거래량을 짧은 주기(일봉 주가를 분봉에서 활용)에 편리하게 적용할 수 있게 되어 있다. 분 차트를 이용한 매매전략에서도 그 다음 긴 주기(일봉)의 가격을 참조하는 방법은 매매 전략을 작성할 때 자주 사용되는 방법으로서 <표 3-1>의 내용을 기억하기 바란다.

그 다음으로 주가 변화의 크기인 변동성(volatility)을 많이 이용하게 되는데, 주로 ATR, Range 등이 사용된다. 전략을 개발할 때 자주 사용되는 개념으로서 필히 알아두어야 한다.

(1) 변동성(Volatility) 측정 지표

변동성이란 주가의 상승, 하락 방향성에 관계 없이 진폭만을 고려한 것으로 옵션에서는 아주 중요한 변수로 사용된다. 시스템 트레이딩에서도 자주 사용되며, 매매 전략을 가격 변동에 따라 변화되게끔 만들어주는 요소이다. 즉 변동성을 고려하게 되면 일률적인 적용에서 벗어나게 되며, 변동성의 크기에 따라 가변적(adaptive)으로 변화시킬 수 있게 된다.

가장 기초적으로는 고가(high) - 저가(low)의 거리를 이용한 Range 가 있다.

1) Range

레인지(Range) 는 [고가-저가]를 말한다. 주로 이동평균 시킨 값을 사용하게 되는데, CT언어로 표시하면 다음과 같다.

〈수식 0-1〉 Range 단순 이동평균식

CT 영역: 지표

이름: Arange(Average Range)

Input: len(10)

Var1=high-low ‘고가 - 저가를 var1에 저장

Var2=mov(var1, len, s) ‘var1을 10일 단순 이동평균 시켜서 var2 에 저장

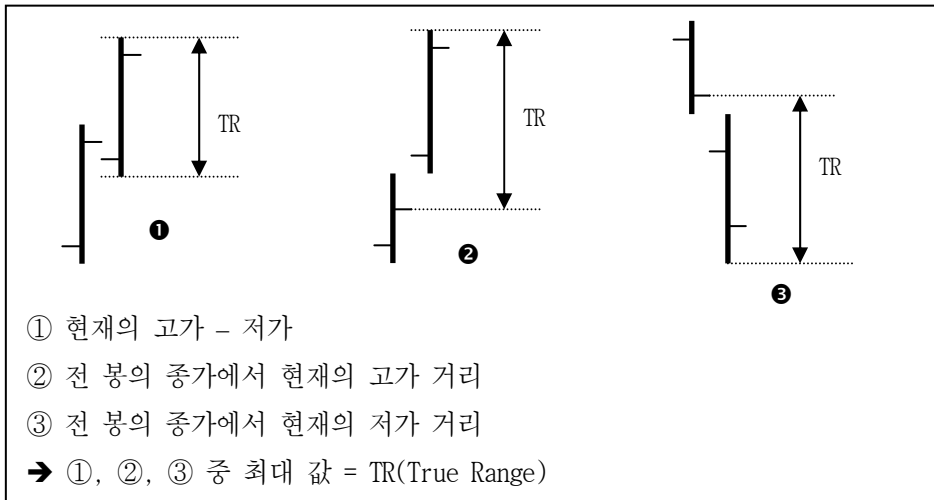
Call plot1(“Arange” , var2) ‘var2를 지표로 표현

레인지값이 크면 변동성이 크다는 의미이고, 값이 적으면 변동성 적다는 의미가 된다. 그 다음으로 더욱더 유용하게 사용되는 변동성 측정 지표로서는 TR(true range)과 ATR(average true range)이 있다.

2) TR, ATR

ATR은 TR을 이동평균하는 개념으로서 TR을 먼저 이해해 보자. TR은 두 봉의 최대 거리를 표현하는 것이다.

〈그림〉 0-1 TR



주) 관련 책자 및 저자: “New concepts in technical trading systems”, Welles j. Wilder, JR.

개발자(Wilder)는 TR에서 ATR을 계산할 때 지수 이동평균법(Exponential moving average method)과 유사한 방법을 사용하였다. 따라서 사이보스 트레이더에서는 원전에서의 계산방식을 그대로 사용하였으나, TR을 단순 이동평균하여 ATR을 계산하는 방법을 사용해도 무방하다. 이 2가지 방법은 다음과 같이 코딩할 수 있다.

〈수식 0-2〉 ATR(Wilder방식)

CT 영역: 지표

이름: ATR(Average True Range)

Input: len(10)

Var1=max(high, close(1)) - min(low, close(1)) ' TR 계산, var1에 저장

Var2=(var2(1)*(len-1)+var1)/len '고전적인 방법으로 ATR 계산, var2 에 저장

Call ploti1(“ATR” , var2) 'var2를 지표로 표현

〈수식 0-3〉 ATR(단순이평방식)

CT 영역: 지표

이름: ATRS(Average True Range Simple)

Input: len(10)

Var1=max(high, close(1)) - min(low, close(1)) ' TR 계산, var1에 저장
 Var2=mov(var1, len, s) ' var1을 10일 단순 이동평균 시켜서 var2 에 저장→ ATRS
 Call ploti1("ATRS" , var2) 'var2를 지표로 표현

ATR 이나 ATRS 역시 값이 크면 변동성이 크다는 것을 의미하고, 적으면 변동성이 낮은 상태라는 것을 나타낸다. 그 다음으로 옵션에서는 아주 중요하게 사용되는 역사적 변동성(historical volatility) 이 있다.

3) 역사적 변동성(historical volatility)

역사적 변동성은 자연로그를 취한 종가 변화의 표준편차를 이용해서 변동성을 측정하는 방법이다. 주로 일간 차트에서 활용되는데 식은 다음과 같다.

〈수식 0-4〉 역사적 변동성

CT 영역: 지표

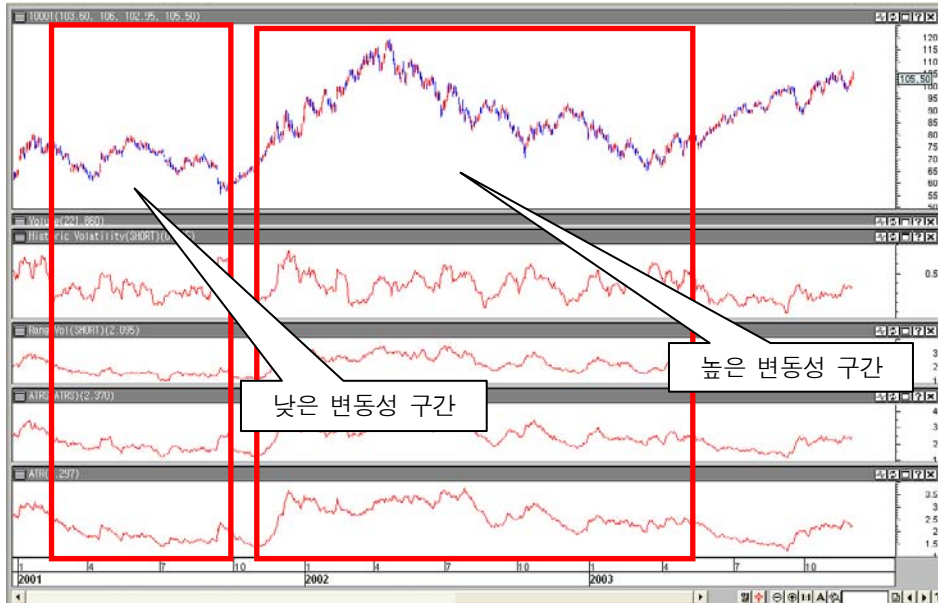
이름: Historical Volatility

Input: len(10)

If close(1)>0 And close>0 Then ' 0으로 나누는 오류 방지
 Var3=log(close/close(1)) ' 로그 수익률 계산
 End If
 Var4=stdev(Var3, len, 1)*sqr(365) '표준편차 $\times \sqrt{365}$
 Call ploti1("historical volatility", Var4)

역사적 변동성 역시 값이 크면 변동성이 크다는 의미이고, 값이 작으면 변동성이 작다는 의미이다. 여기서 Range, ATR, Historical Volatility를 일간 연결 선물 지수에 적용하여 변동성의 의미를 파악해보기로 하자.

〈그림 0-2〉 변동성 측정지표



주) 기간 값: 10일, 연결선물지수 일간 차트

위 그림을 보면 모두다 가격 변동성을 잘 표현해주고 있다. 즉 주가가 상승하거나 하락하는 것에 관계없이 가격 진폭이 증가하면 지표들 값이 상승하고, 진폭이 감소하면 하락하는 특징을 나타내고 있다.

이중 Range, ATR, ATRS의 흐름은 유사함을 알 수 있다. Historical Volatility의 경우 급변 모형(sharp move)이 남아 있는 형태로서 시스템 트레이딩에서 변동성을 추세적으로 판단하기 위해서는 잘 사용되지 않는다. 가장 일반적으로 사용하는 지표는 ATR 이거나 ATRS 이다.

그 다음으로 이 책자에서는 대부분의 전략이 앞서 설명한 주문유형 중 Atstop 주문 유형으로 되어있는데 그 이유에 대해 알아보자.

(2) Atstop 주문의 유용성

CT 주문 유형은 크게 Atmarket주문과 Atstop주문, Atlimit주문으로 구분된다.

이중에서 가장 유용하게 사용되는 주문 유형이 Atstop주문 유형이다.

우선 Atlimit주문과 Atstop주문 유형을 비교해보면, 매수의 경우 Atlimit는 지정가 주문(일정가격 이하이면 매수)으로서 역추세적인 진입 방법에서 자주 사용되는 주문 형태이다. Atstop은 “일정가격 이상이면 매수 하라”라는 확인매수 형태이다. 매매 전략에 따라 차이는 있지만, 확인매매인 Atstop 형태가 시스템 효율성 증대 및 테스트 결과를 향상시키는 장점이 있어 Atstop 주문 형태가 좀더 유용하다고 할 수 있다.

1) Atstop과 Atmarket 주문의 비교

〈수식 0-5〉 Atstop 주문의 유용성

영역: 전략
이름: StopAtmarket_Test
Input: len(60), method(1)
<pre>If method=1 Then ‘ method 가 1 이면 If crossup(close, mov(close, len, s)) Then ‘종가가 이동평균선을 상향 돌파하면 Call buy("매수", Atmarket) ‘ 그 다음봉에 시장가 매수 하라 End If If crossdn(close, mov(close, len, s)) Then ‘ 종가가 이동평균선을 하향돌파하면 Call sell("매도", Atmarket) ‘ 그 다음봉에 시장가 매도하라 End If Else ‘ method 가 1 이 아니면 If close(1)>mov(close, len, s, 1) Then ‘ 전봉 종가가 전봉 이동평균선 보다 크면 Call buy("매수", Atstop, Def, high) ‘ 현재 봉 고가를 다음 봉이 돌파하는 시점에서 End If ‘ 매수 하라 If close(1)<mov(close, len, s, 1) Then ‘ 전봉 종가가 전봉이동평균선 보다 적으면 Call sell("매도", Atstop, Def, low) ‘ 현재 봉 저가를 다음 봉이 하향 돌파하는 End If ‘ 시점에서 매도하라. End If</pre>

예를 들어, “주가가 이동평균선을 상향돌파하면 매수하라”가 Atmarket 주문

형태이고 “주가가 이동평균선보다 커지면 돌파시점 이후의 고가를 돌파할 경우에만 매수하라”가 Atstop 주문 형태이다.

<수식 3-5>를 보면 method를 1로 입력하면 Atmarket주문 형태를 테스트 하게 되며 method를 1이외의 숫자를 입력하면 Atstop 형태의 주문을 테스트 하게 된다. 실제로 일간 연결선물지수를 이용해서 두 가지 주문 형태에 대해 테스트를 해보자

<그림 0-3> Atmarket 주문일 경우



주) 연결선물지수 일간 차트: 이동평균선 60일, method: 1

<그림 0-4> Atstop 주문일 경우



주) 연결선물지수 일간 차트: 이동평균선 60일, method: 2

<그림 3-4>을 보면 <그림 3-3>과 다르게 매수주문이 종가가 이동평균선보다 커진 뒤에 다시 가격이 고가 돌파가 발생하면 매수하는 것을 알 수 있다. 반대의 경우에는 매도가 발생한다. 따라서 <그림 3-3>에서 발생하던 반복적인 속임수 매매가 제어됨을 알 수 있다. 이 두 가지 주문 방법의 테스트 결과는 다음과 같다.

<표 0-2> 테스트 결과 비교 분석

평가 항목	Atmarket	비 교	Atstop
총 손익	52,322.30	<	70,558.91
평균 손익	545.02	<	1,720.95
최대 손실	-4,607.13	<	-4,860.01
총 매매수	96	>	41
연속 이익 매매수	3	=	3
연속 손실 매매수	12	>	4
최대 자본 인하액	-20,454.44	<	-13,929.59
승률	23.96	<	41.46
손익비	1.55	<	2.30
평균손익비	4.92	>	3.25
보상 비율	2.56	<	5.07

주) 단위: 천원, 테스트 기간: 1996. 04. 15 ~ 2003. 12. 02, 필수적으로 보유할 바 수: 50

<표 3-2>를 보면 전체적으로 Atstop이 속임수 매매가 적으면서도 더 많은 수익을 발생한 것으로 분석되어 Atstop 이 Atmarket 보다 유용한 주문 형태 라는 것을 알 수 있다(더 나아가 이동평균20~100일까지 최적화를 한 결과에서도 Atstop 주문이 유용함을 알 수 있었다).

결론적으로 Atstop 주문 형태가 자체적으로 속임수 매매를 제어하는 필터를 포함하게 되고, 봉 중간에 진입하는 즉시성을 가지고 있는 유용한 주문 방법임을 알 수 있을 것이다. 전문적으로 Atstop 주문 형태를 “Setup “과 “Entry”로 구분하는 방법이라 한다.

2) Setup 과 Entry

셋업(Setup)과 진입(Entry) 방법은 특히 기술적 지표를 이용한 매매전략에서 매매신호(event)가 발생하면 바로 진입하는 것이 아니라, 이를 셋업으로 정의하고 실제 주문은 가격으로 확인하는 방법을 말한다. 예를 들어 “스토케스틱이 신호선을 돌파하면 Setup 상태가 되고, 진입(Entry)은 3봉 고가를 상향 돌파하면 매수”와 같이 사용하는 것을 말한다.

Charlie F. Wright의 “Trading as a Business”에서 이러한 부분을 자세히 설명하고 있는데, 셋업과 진입에 대하여 몇 가지 정리해보면 다음과 같다.

■ 셋업과 진입의 종류(매수 경우)

셋 업	진 입
단기 이동평균선이 장기 이동평균선보다 커지면	현재 고가를 다음 봉이 돌파하면 매수
ADX 지표가 상승 추세이면	3봉 고가를 다음봉이 돌파하면 매수
주가가 주가 채널을 벗어나기 시작하면	3봉 저가가 현재 저가라면 다음 봉이 현재 고가를 돌파하면 매수
RSI, 스토케스틱 지표가 과매도(30) 구간에 들어가면	swing high 포인트를 상향 돌파하면 매수
상승 갭이 전일 고가 이상으로 발생하면	3봉 연속 증가 상승이 발생하면 다음 봉이 현재 고가를 돌파할 경우에 매수

따라서 이 책에서는 거의 모든 전략이 셋업과 진입으로 구분되며, 주문 유형은 Atstop 방법을 사용할 것이다.

이 이외에도 시간 함수 및 주문 횟수 제어, 전략 함수 사용 방법 등 기초적으로 알아야 할 부분들이 많지만 이러한 부분들은 실제로 따라 하기를 진행하면서 이해하도록 하자.

여기까지가 전략 개발하기 전에 알아두어야 할 내용들이다. CT 가격 함수 및 변동성, Atstop 주문의 유용성 등은 앞으로 전략을 개발할 때 필요한 기초 지식이 될 것이다. 그러면 이제부터 데이트레이딩 전략을 개발해 보도록 하자.

2. 데이트레이딩

Jake Bernstein 의 “The Complete Day Trader”에서는 데이트레이딩의 장점에 대해 다음과 같이 기술하였다.

왜 데이트레이드 인가? (Why Day Trade?)
<ul style="list-style-type: none">■ 자본을 최대한 활용할 수 있다(maximized equity): 오버나잇을 하면 마진콜 등에 대비한 추가 증거금을 설정해야 하는데, 데이트레이딩은 이러한 오버나잇 위험이 없어 자본을 최대한 활용할 수 있다.■ 위험 노출을 줄일 수 있다(reduced risk exposure): 그 다음날 시가갭의 위험에 노출될 위험을 없앨 수 있다.■ 강제 청산의 유용성(advantage of forced exit): 50% 승률이라도 오버나잇의 경우 정확한 가격에 청산을 할 수가 없다. 하지만 연속적인 데이터들(open gap이 없는) 사용하는 day trading은 빠르고 정확한 가격 청산이 가능하다.■ 신호의 신뢰성(reliability of timing signals): 많은 시스템 트레이딩 신호는 긴 시간대보다 짧은 시간대에서 더 신뢰성이 크다.■ 빠른 회수가 가능하다(immediate feedback): 수익획득이나 손실이 빠른 시간내에 확정되며, 매매전략의 수정 또한 빠르게 진행 할 수 있다.■ 빠른 개발이 가능하다(faster development time): 포지션 트레이딩 보다 빠른 개발과 테스트가 가능하다.

제 2 장에서 설명한 바와 같이 데이트레이딩은 하루 중에 매매 포지션을 완전히 청산하는 방법을 사용한다. 그 이유도 설명하였는데 매매 포지션을 그 다음날로 가져 간다는 것은 오버나잇 위험(시가갭)을 감수하겠다는 것으로서, 데이트레이딩에서는 오로지 장 중 위험만을 감수하고 장중 수익만을 획득하는 것이 목적이므로 절대 허용되지 않는다.

선물이나 옵션 등 파생 상품 매매에서는 데이트레이딩이 일반적이다. 그 이유는 역시 오버나잇 위험이 크기 때문이다.

현물 시장에서는 미수가 발생하지 않는 경우에 오버나잇 이후 손실이 발생한다면 오랜 기간 동안 보유(holding)하는 경우로 전환되어도 무방하다. 하지만 선물, 옵션 시장은 매매 증거금(margin)이 낮아 레버리지를 높여 수익 극대화를 도모하는데, 당일 결제이기 때문에 오버나잇을 하여 손실이 발생하게 되면, 마진콜(margin call)이 발생하여 반대매매를 당할 수 있기 때문이다.

여기서 선물/옵션의 수익 구조에 대해 간단한 예를 통해서 살펴보자. 현재 선물이 100포인트라 가정하자.

상승을 기대하고 1 계약을 매수할 경우 필요한 증거금은 $100 \times 1 \times 500,000 \times 0.15 = 7,500,000$ 원이 된다. 그런데 초기 증거금이 현재 규정상 15,000,000 원이 필요하므로 15,000,000 원으로 1 계약을 100 포인트에 매수하게 된다. 이때 실제로 매수한 금액 50,000,000 원이다.

다음날 선물가격이 5% 상승하여 105 포인트가 되었다. 따라서 105에 매수청산을 하였다면 매매금액은 52,500,000 원이 되어, 수익은 $52,500,000 - 50,000,000 = 2,500,000$ 원이 발생한다. 투자 금액 대비 수익률은 $2,500,000 / 15,000,000 \times 100 = 16.67\%$ 로 선물가격이 5% 상승함에 따라 투자금액 대비 수익은 16.67% 발생하게 된다. 이러한 레버리지 효과로 인해 선물매매는 상당히 매력적인 투자 방법이 된다.

반대의 경우를 생각해보자. 그 다음날 5% 하락하였다. 그러면 95 포인트가 되고 손실은 $50,000,000 - 47,500,000 = 2,500,000$ 원이 되며 투자 금액 대비로 보면 $2,500,000 / 15,000,000$ 원으로 16.67% 손실이 발생하게 된다. 따라서 매수 청산을 하였다면 12,500,000 원이 투자금액이 되어 추가 자금이 필요하게 된다.

옵션의 경우는 더하다. 주가지수 옵션의 경우 그 다음날 투자금액에 2~3 배 수익이 발생할 수도 있고, 50 ~ 80% 이상 손실이 발생할 수 있다.

따라서 장중 매매로 충분히 수익이 발생할 수 있는 구조를 가지고 있는 것이

선물, 옵션 시장이기 때문에 투자자금이 적거나, 투자 경험이 적은 투자자뿐만 아니라 전문투자자들도 데이트레이딩을 선호하게 된다.

데이트레이딩은 장중 변동 및 방향성을 이용해서 수익을 발생시키는 매매 방법이다. 장중 변동성이 크고 방향성도 한 방향으로 진행하면, 추세적인 매매 방법으로 수익을 발생시킬 수 있다.

반대로 장중 변동성이 크고 추세가 없으면, 역추세적인 매매 전략이 수익을 발생시킬 수 있다. 그러나 장중 변동성이 적으면, 수익 폭이 적거나 손실이 발생할 확률이 크므로 매매를 자제하는 것이 바람직하다.

<수식 0-6> 장중 추세 분석 지표

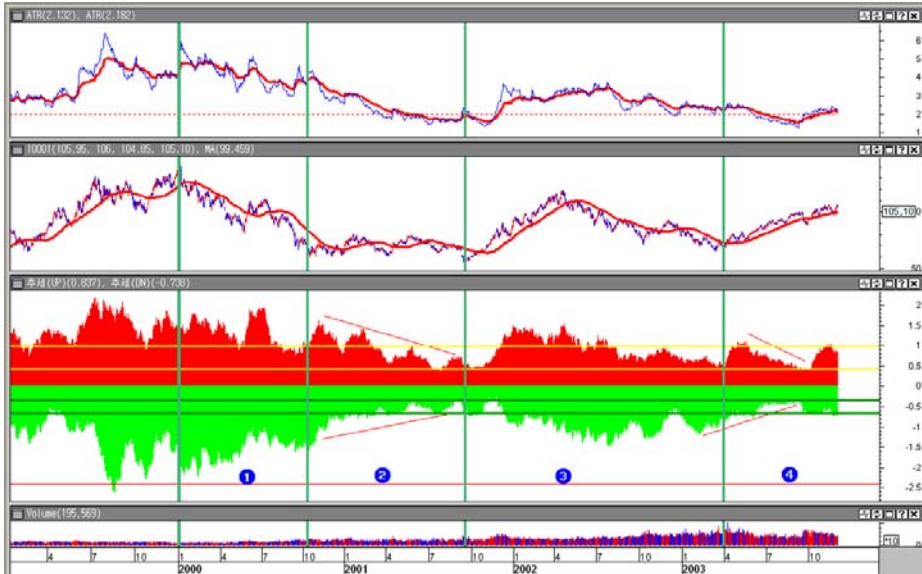
영역: 지표	
이름: 추세	
Input: len(10)	
If open<=close Then	‘종가가 시가보다 크거나 같으면(양봉이면)
Var30 = high-open	‘고가-시가를 var30에 할당
Var31=0	‘var31에는 0을 할당
Else	‘종가가 시가보다 작으면(음봉이면)
Var31=low-open	‘저가 - 시가를 var31에 입력
Var30=0	‘var30 에는 0을 입력
End If	
Call plot1("up", mov(Var30, len, s))	‘var30의 단순 이동평균선을 표시
Call plot2("dn", mov(Var31, len, s))	‘var31의 단순 이동평균선을 표시

그러면 선물 가격은 변동성과 어떤 관계인가? 앞서 설명한 바와 같이 변동성을 분석하는 지표는 ATR(average true range)를 보면 알 수 있고, 추세는 이동평균선을 통해서 파악할 수 있을 것이다. 추가로 장중 추세를 분석하기 위해서 장 중 수익구조 지표를 만들어서 분석해보자

<수식 3-6> 는 장 시작할 때 시가를 기준으로 양봉이면 시가에서 고가까지를 매수 힘으로 판단하고, 음봉이면 시가에서 저가까지를 매도 힘으로 판단하여 표시

하는 지표이다. 그러면 일간 연결 선물 지수에 표시를 하여 분석을 해보자

〈그림 0-5〉 과거 변동성 및 추세 분석



주) 연결선물 지수 일간 차트, 분석기간: 2001. 1 ~ 2003. 12

〈그림 3-5〉 는 제일 위에 ATR(10)과 ATR(30)을 동시에 표시하였다(굵은 선이 30) 중간에는 연결선물지수에 50일 단순 이동평균선을 표시하였다. 세 번째는 장중 추세 지표를 적용하였다.

0선 위는 시가에서 고가까지를 나타내고 0선 아래는 시가에서 저가까지를 나타낸다. CT에서 제공하는 분데이터는 2000년 1월 4일부터 이므로 2000년 이후를 분석해보면 그림에 표시된 것과 같이 ①, ②, ③, ④번 구간으로 나누어질 수 있다.

〈표 0-3〉 과거 변동성 및 추세 분석

구간	ATR	이동평균선	장중 추세 지표
①	>2, 높은 수준	하락 추세	높은 수준
②	<2, 낮은 수준	추세 없음	낮은 수준
③	>2, 높은 수준	상승 및 하락추세	높은 수준
④	<2, 낮은 수준	상승 추세	낮은 수준

①, ③구간은 추세가 있으면서 변동성도 크고, 장중 추세도 큰 상황으로서 데이트레이딩이나 단기매매에서 수익이 발생할 것을 예상할 수 있다. 반면 ②번 구간은 추세가 없고 장중 변동성도 낮은 수준으로서, 단기매매나 데이트레이딩으로서 수익이 발생할 가능성이 적은 구간임을 알 수 있다.

④번 구간은 2003년 4 월 이후 상승추세 구간임에도, 변동성 및 장중 추세가 낮은 수준의 구간이다. 즉 상승추세로서 오버나잇을 통한 수익 창출은 가능하지만, 데이트레이딩으로는 수익을 발생하기 힘든 특이한 구간임을 알 수 있다 (2003년 10월 이후에는 상승추세와 변동성, 장중 추세가 다시 증가하고 있어 데이트레이딩 가능 구간으로 바뀌고 있다).

따라서 데이트레이딩 전략을 테스트하면 ②, ④ 번 구간에서 수익이 발생하지 않고 드로다운(연속손실)이 발생할 것이라 것을 예상할 수 있다. 특히 일반적으로 범용적(robust)인 데이트레이딩 전략은 대부분 돌파 전략이거나, 변동성 확대(volatility expansion) 전략이다. 즉, 추세가 발달하는 방향으로 매매하는 추세추종적인 형태의 전략이므로, ②, ④ 번 구간에서 수익 감소가 발생할 것이다.

이러한 현상을 잘 기억해야 한다. 왜냐하면 전략을 개발하다 보면 ②, ④번 구간에서 수익이 발생하는 매매 알고리즘을 찾을 수 있을 것이고, 그렇게 된다면 기존의 매매 전략에 결합하여 상당히 시스템 효율성이 향상될 것이기 때문이다.

간단하게 각 구간별 대응 전략을 정리하고 이제부터 본격적으로 데이트레이딩 전략을 작성해보자.

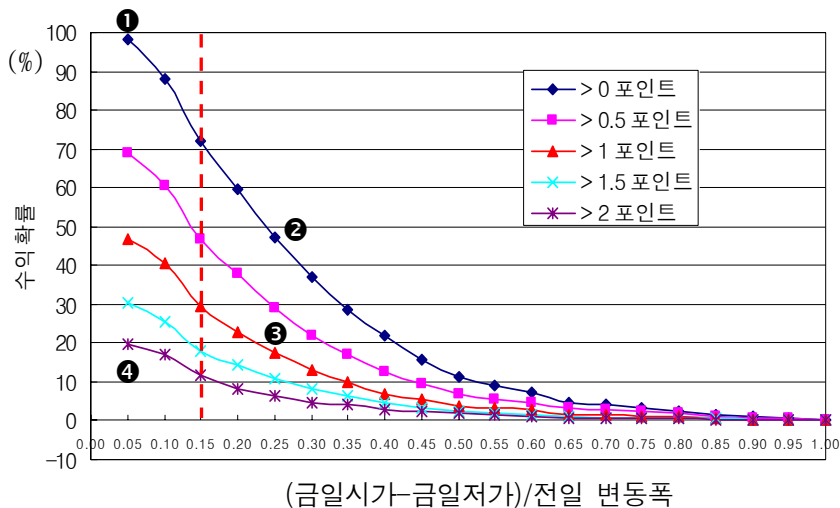
〈표 0-4〉 각 구간별 대응 전략

구간	대응전략
①	숏텀, 데이트레이딩, 추세추종적인 돌파 전략
②	데이트레이딩만 사용, 역 추세적인 매매 형태 적합
③	숏텀, 데이트레이딩, 추세추종적인 돌파 전략
④	숏텀 트레이딩과 데이트레이딩 전략을 동시에 사용 추세추종적인 돌파 전략에 역 추세적인 매매 가미

(1) 진입(Entry)

시장에 진입하는 의사 결정은 대부분 가격이나 거래량, 지지 및 저항선, 기술적 지표 등에 의한다. 다양한 진입 방법은 4장에서 다루기로 하자. 여기서는 시가, 종가의 관계와 전일의 Range(high-low)간의 관계를 이용한 전략을 알아보자.

〈그림 0-6〉 매수의 경우



〈그림 3-6〉은 금일 매수포지션을 취할 때 전일 변동폭(전일고가-전일저가) 대비 금일의 매도힘(금일시가-금일저가)의 비율(x축)에 따라 발생 가능한 수익별 확률(y축)을 표현한 그래프이다.

수익규모에 따라 5가지로 나누었다. ◆은 0포인트이상, ■는 0.5포인트이상, ▲는 1포인트이상, ×는 1.5포인트이상, *은 2포인트이상 수익이 발생한 확률을 의미한다.

그래프의 해석은 다음과 같다.

①은 금일 장 시작시 매수하여 종가청산 할 경우, 금일의 매도힘이 전일 변동폭대비 0.05 미만이면, 0포인트 이상 수익이 발생할 확률은 98%라는 것이다.

②는 금일 장시작시 매수하여 종가청산 할 경우, 금일의 매도힘이 전일 변동폭 대비 0.2 이상 0.25미만인 경우 0포인트 이상 수익이 발생할 확률은 48%라는 것이다.

③은 금일 장시작시 매수하여 종가청산 할 경우, 금일의 매도힘이 전일 변동폭 대비 0.2이상 0.25미만인 경우 1포인트 이상 수익이 발생할 확률은 18%라는 것이다.

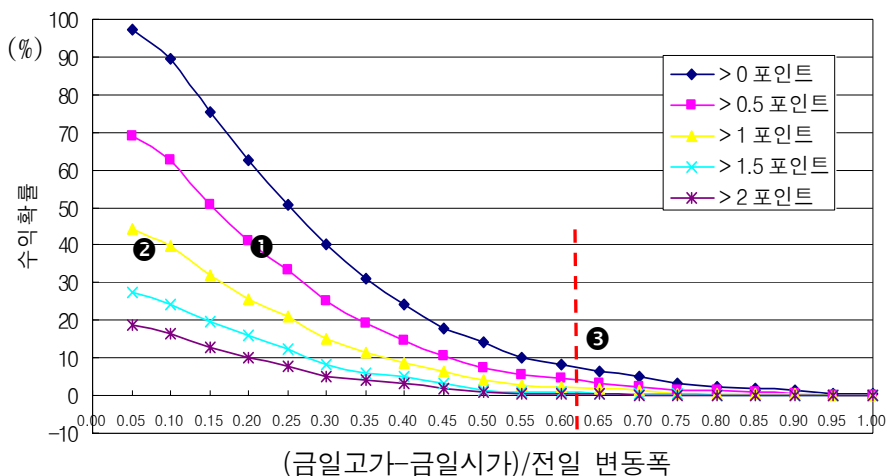
④은 금일 장시작시 매수하여 종가청산 할 경우, 금일의 매도힘이 전일 변동폭 대비 0.05미만인 경우 2포인트 이상 수익이 발생할 확률은 19.5%라는 것이다.

그래프를 종합적으로 보면, [금일매도힘/전일 변동폭] 즉 X축 값이 1에 가까워 질수록 매수시 수익이 발생할 확률이 급격히 줄어든다(<그림 3-6>에서 x축의 값이 1에 가까워질수록 수익확률이 0에 수렴한다). [금일매도힘/전일 변동폭]이 일정할 경우 수익이 크게 발생할 확률은 현저히 감소한다(수직선).

즉, 금일 시가대비 금일저가가 전일의 변동폭의 일정수준 이상 하락하는 경우 (매도힘이 전일 변동폭 대비 커질수록) 금일 종가는 금일 시가 이상에서 형성되기 힘들다(매수 후 수익실현이 어렵다)는 것을 알 수 있다.

매도의 경우도 살펴보자.

<그림 0-7> 매도의 경우



<그림 3-7>은 매도의 경우로서 발생가능 한 수익별 확률(y축)을 전일변동폭 대비 매수힘(금일고가-금일시가)의 비율(x축)에 따라 표현한 그래프이다.

①은 금일 장 시작시 매도하여 증가청산 할 경우, 금일의 매수힘이 전일변동폭 대비 0.15이상 0.2미만인 경우 0.5포인트 이상 수익이 발생할 확률은 41%라는 것이다.

②는 금일 장 시작시 매도하여 증가청산 할 경우, 금일의 매수힘이 전일 변동폭 대비 0.05미만인 경우 1포인트 이상 수익이 발생할 확률은 44%라는 것이다.

③은 금일 장 시작시 매도하여 증가청산 할 경우, 금일의 매수힘이 전일 변동폭 대비 0.55이상 0.6미만인 경우 0, 0.5, 1, 1.5, 2포인트 이상 수익이 발생할 확률(점선)은 모두 10%를 넘지 못한다는 것이다.

매수의 경우와 같이 [금일 매수힘/전일변동폭]이 1에 가까워 질수록 매도시 수익이 발생할 확률은 급격히 줄어들며, [금일매수힘/전일변동폭]이 일정할 경우 수익이 크게 발생할 확률은 현격히 작아진다.

즉, 금일 시가대비 금일 고가가 전일 변동폭의 일정수준이상 상승하는 경우 (매수힘이 전일 변동폭 대비 커질수록) 금일 증가는 금일 시가 미만으로 형성되기 힘들다(매도후 수익실현이 어렵다)는 것을 알 수 있다.

수익이 1 포인트 이상 발생할 확률만 가지고 표를 만들어 보면 다음과 같다.

〈표 0-5〉 수익이 1포인트 이상 발생할 확률

매 수			매 도		
시가-저가/ 변동폭	건수	수익확률	고가-시가/ 변동폭	건수	수익확률
0.05	58	46.99	0.05	43	44.51
0.1	97	40.39	0.1	66	39.54
0.15	58	29.35	0.15	54	31.91
0.2	48	22.75	0.2	40	25.66
0.25	38	17.29	0.25	51	21.04
0.3	25	12.97	0.3	32	15.14
0.35	27	10.13	0.35	22	11.45
0.4	13	7.05	0.4	23	8.90
0.45	15	5.57	0.45	20	6.24
0.5	6	3.87	0.5	9	3.93
0.55	4	3.19	0.55	4	2.89
0.6	9	2.73	0.6	5	2.43
0.65	3	1.71	0.65	5	1.85
0.7	2	1.37	0.7	5	1.27
0.75	1	1.14	0.75	1	0.69
0.8	4	1.02	0.8	0	0.58
0.85	2	0.57	0.85	2	0.58
0.9	0	0.34	0.9	2	0.35
0.95	2	0.34	0.95	1	0.12
1	1	0.11	1	0	0.00

위 표를 보면 (금일시가-금일저가)/전일변동폭이 0.35를 초과하면 매수해서 수익이 1포인트 발생할 확률이 10% 미만이며, 금일고가-금일시가/전일변동폭이 0.35를 초과하면 매도해서 1포인트 이상 수익이 발생할 확률이 10% 미만이 된다.

“Long-Term Secrets to Short-Term Trading”의 저자 Larry William은 이런 분석을 통해서 다음과 같은 결론을 도출하였다.

- ① 종가가 상승할 것으로 기대해서 시가보다 아주 낮은 가격에서 매수하려고 하지 마라(Don' t try to buy dips below the open on expected up close days).
- ② 매수 상태에서 시가보다 많이 하락하면 청산하라(If long and price fall much below the open on expected big up close days, " Get out").
- ③ 종가가 하락할 것으로 기대해서 시가보다 아주 높은 가격에서 매도하려고 하지 마라(Don' t try to sell big rally much above opening on expected large down days).
- ④ 매도 상태에서 시가보다 많이 상승하면 청산하라(If short and price rally much above opening on expected large down days, " get out").

이러한 원리를 근거로 Larry William, Pruitt 등 많은 전략 개발자들이 제시하는 매매 전략은 시가 돌파 전략이다(예: Open-Range Breakout, Range Break, Volatility Breakout 등).

1) Range Break 전략

앞서 설명한 부분을 다른 의미로 설명하면 다음과 같다. 시가 대비 전일 변동 폭(Range)의 일정 수준이상 상승하면 종가가 시가 아래로 하락하기 힘들고, 반대로 시가 대비 전일 range의 일정 수준이상 하락하면 종가가 시가 이상이 되기가 힘들다는 이야기가 된다. 따라서 시가 대비 전일 Range의 일정 수준 이상을 돌파하면 매수, 시가 대비 전일 Range의 일정 수준 이하를 하향 돌파하면 매도하는 돌파 전략을 만들 수 있다. CT를 이용해서 전략을 만들어 보자

〈수식 0-7〉 데이트레이딩 전략(단계 1)

영역: 전략

이름: DD_Rangebreak(v 0.1)

Input: len(0.3)

Var1= highd(1)-lowd(1) '전일 Range를 var1에 입력

Call buy("매수", Atstop, Def, opend+var1*len) '금일 시가 + range * len 돌파 시 매수

Call sell("매도", Atstop, Def, opend-var1*len) '금일 시가 - range * len 돌파 시 매도

<수식 3-7>은 가장 기초적인 식이다. 위 식과 같이 작성할 경우, 데이트레이딩이기 때문에 종가 청산을 적용해야 한다. 종가 청산은 2장에서 설명한 것처럼 동시호가에는 청산 주문을 할 수 없다. 동시호가 바로 직전에 가능하다. 방법은 두 가지이다. 첫째는 전략 속성창에서 선택하는 방법과 코딩으로 제어하는 방법이다.

①속성창에서 당일 청산 적용하기

<그림 0-8> 속성창에서 종가 청산 설정

전략을 연결선물지수 분간 차트에 적용하고 과거 테스트를 할 때, 속성 설정에서 두 가지는 무조건 결정해주어야 하는데, 첫째는 슬리피지(Slippage)이다. 선물의 경우 적어도 1틱(0.05 포인트)은 설정해주어야 하며, 필수적으로 보유해야 할 바 수를 넉넉하게 설정해주어야 한다(제 2장 참조).

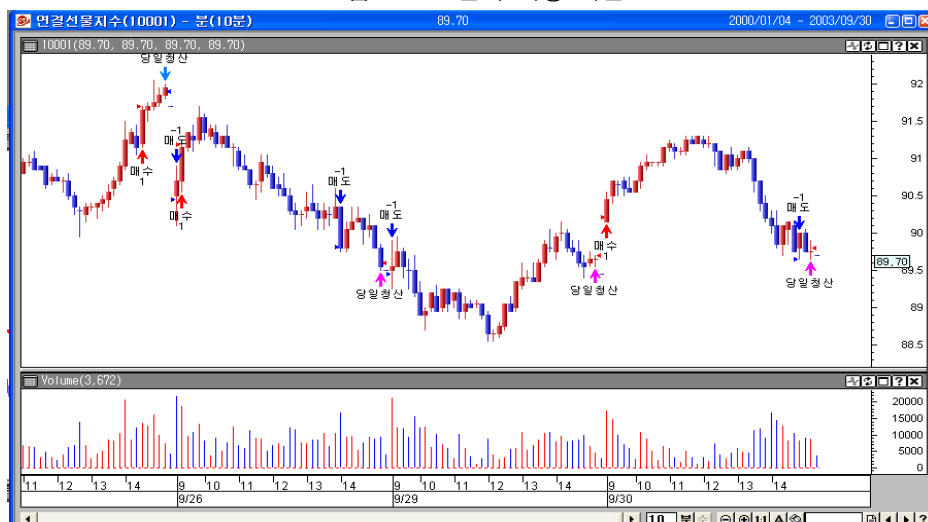
그 다음으로 데이트레이딩일 경우 종가 청산을 선택해서 시간을 입력하면 종가 청산이 작동하게

된다. 동시호가에 청산을 할 수 없으므로 동시호가 시작 바로 전인 15:04 분으로 설정하면 된다. 우선 다음과 같은 순서로 속성창 설정까지 진행한다.

- 메뉴에서 [설정] → [모드 변환] → [오프라인 모드]로 설정하든지 처음부터 오프라인으로 접속한다.
- 새로운 워크 그룹을 생성하고, 연결 선물 지수 10분 차트를 생성한다.
- 데이터 로드는 2000년 1월 4일부터 ~ 현재까지(이 책에서는 일관성을 위해서 2003년 9월 30일로 제한함)
- 전략 “RangeBreak(v0.1)을 적용한다.
- 이때 적용창의 “비용” 탭에서 슬리피지 적용방식을 가격에 반영, 슬리피지를 최소 변동 단위 선택과 “1”을 입력하여 0.05 포인트 슬리피지가 작동되도록 한다.
- “기본 설정”에서 필수적으로 보유해야 할 바 수를 50(이 전략의 경우 1 도 무방함)으로 설정하고 1 계약으로 매매하도록 한다.
- 주문 설정은 시험 적용으로 한다.
- 당일 청산을 선택하고 시간을 15:04 분으로 선택한다.

여기서 확인을 클릭하면 차트에 적용된다. 적용된 화면은 다음과 같다.

〈그림 0-9〉 전략 적용 화면



주)10분 선물 차트, 기간: 2000. 01. 04 ~ 2003. 09. 30

시스템 개발을 해본 투자가의 경우 현재 코딩 오류가 포함되어 있다는 것을 바로 알 수 있지만, 초보자의 경우에는 파악하기 힘들 것 이다. 우선 코딩 오류 등을 체크 하기 전에 전략 평가를 간단하게 해 보자(“시스템 트레이딩” → “시뮬레이션 보고서”).

〈표 0-6〉 테스트 결과

평가 항목	전 체	매 수	매 도
총 손익	42,869.83	13,049.06	29,820.77
평균 손익	28.43	17.15	39.92
최대 손실	-4,263.38	-4,263.38	-1,529.01
총 매매수	1,508	761	747
연속 이익 매매수	7	5	8
연속 손실 매매수	10	10	10
최대 자본 인하액	-9,716.85	-11,826.21	-13,605.59
승률	43.50	44.15	42.84
손익비	1.12	1.07	1.17
평균손익비	1.46	1.36	1.57
보상 비율	4.41	1.10	2.19

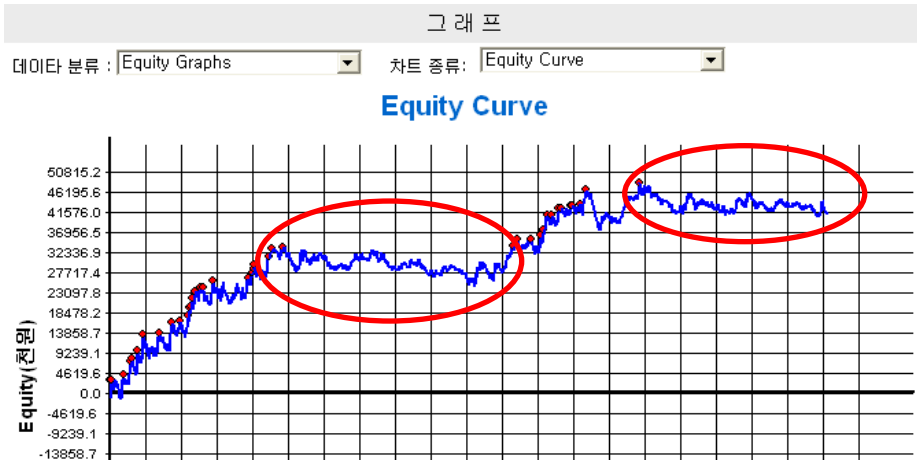
주) 단위: 천원, 테스트기간: 2000. 01. 04 ~ 2003. 09. 30,
필수적으로 보유해야 할 바수: 50, 슬리피지: 1틱

<표 3-6>을 보면 전체적으로 수익 및 위험의 구조가 나쁜 편은 아니라는 것을 알 수 있다. 즉 추가 개발이 가능하다는 판단을 할 수 있다.

실제로 전략을 개발하는 첫 단계에서 진입과 기초 청산 만으로 너무 나쁜 수익 구조를 가지고 있다면, 개발 지속을 생각하기 보다는 새로운 접근 방법을 사용하는 것이 바람직하다.

이제 수익곡선을 보자.

<그림 0-10> 누적수익곡선



누적 수익곡선을 보면 현재까지 (-)수익으로 진행되지 않으므로 나쁘지는 않다. 하지만 원호로 표시된 부분처럼 드로다운(연속 손실)이 상당기간 진행하는 구간이 있음을 알 수 있다. 즉 2001년도나 2003 년부터 테스트를 한다면 총손익이 (-)를 기록할 것이라는 알 수 있다.

역시 이 구간은 <그림 3-5>의 ②, ④ 번 구간으로서 앞서 지적했던 바와 같이 돌과 전략이 수익이 발생하지 않는 구간임을 알 수 있다 (② 번 구간 비추세, 낮은 변동성, ④ 번 구간 상승추세, 낮은 변동성).

종가 청산을 이용하는 방법으로 15:04 분만 사용할 수 있을까? 아니다. 15:00 분도 가능하며 14:30 분도 가능하다.

장 종반에 근접할수록 장중 추세와 반대로 움직이는 경향이 강해진다. 그 이유는 수익을 획득한 투자자가 장 종료시점에서 수익을 실현하기 때문이다. 따라서 동시 호가 바로 전이 당일 청산 시점이 맞지만, 실제적으로는 14:30 이후에는 포지션을 정리하는 당일 청산도 유용한 방법이 된다.

②코딩을 통한 종가 청산

<수식 0-8> 데이트레이딩 전략(단계 1)

영역: 전략

이름: DD_openrangebreakout(v 0.1)

Input: len(0.3)

Var1= highd(1)-lowd(1) '전일 Range를 var1에 입력

Call buy("매수", Atstop, Def, opend+var1*len) '금일 시가 + range*len 돌파시 매수

Call sell("매도", Atstop, Def, opend-var1*len) '금일 시가-range*len 돌파시 매도

If ttime = 1500 then '시간이 15시이면

Call exitlong("종가청산" , atmarket) '매수 포지션을 청산하라

Call exitshort("종가청산" , atmarket) '매도 포지션을 청산하라

End if

<수식 3-8>에서 15시 00분으로 시간을 주었을 경우 10분 봉에서는 그 다음봉인 15시 10분봉 시가에 종가 청산이 발생한다. 따라서 앞서 설명한 속성창 설정에서 15:04 분으로 설정한 것과 다른 포인트에서 청산이 발생하여 결과가 다르다는 것을 유의하자.

앞으로는 [속성창]의 [당일청산15:04]를 기준으로 진행 할 것이다. 이제 추가 개발을 계속 진행 해보자. 그 다음 단계인 코딩 오류를 체크하고, 새로운 알고리즘을 추가하는 작업을 계속 진행하자.

(2) 코딩 오류 및 알고리즘 추가

우선 자신이 원하는 대로 코딩이 되었는지를 검증하기 위해서는 신호 영역을 이용해서 진입과 청산의 위치가 정확한지, 이상한 진입이 없는지를 검증해야 한다.

그럼 신호 영역으로 매매시점을 표시해서 차트에 같이 표현해보자.

<수식 0-9> 데이트레이딩 전략(단계 2)

영역: 신호

이름: DD_RangeBreak(v 0.1)

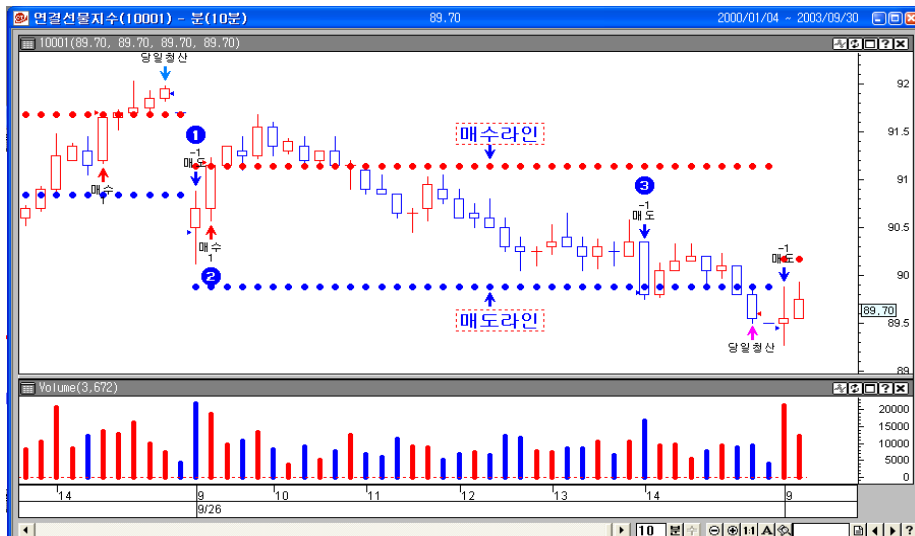
Input: len(0.3)

Var1=highd(1)-lowd(1)

Call plots1("매수라인", open+var1*len)

Call plots2("매도라인", open-var1*len)

<그림 0-11> 신호 및 전략 적용



<그림 3-11>은 매도라인을 하향돌과 할 때 매도하고 매수라인을 상향돌과 할 때 매수하는 전략을 적용한 결과이다. ①②③번 매매 신호를 자세히 보자.

②는 매수라인을 상향돌과하는 시점에서 매수 주문이 발생하여서 원하는 대로 주문이 발생하였고, ③도 매도라인을 하향돌과해서 매도 주문이 정상적으로 발생하였다. 그런데 ①은 매수라인이나 매도라인을 돌과하지도 않았는데 주문이 발생하는 잘못된 매매 신호임을 알 수 있다. 이유는 시간 제어가 되어있지 않아 전일의 매도라인이 유효하기 때문이다.

<수식 3-9>는 매수와 매도가 모든 봉에 활성화 되어 있어 연속적인 작동을 한

다. 하지만 데이트레이딩 전략일 경우에 해당 일에만 작동해야 하는데, 이렇게 해당 일에만 작동하게 하기 위해서는 시간 제어를 해주어야 한다. 즉 “주문은 몇 시까지만 작동하라” 라고 제어해야 하는 것이다.

<수식 0-10> 데이트레이딩 전략(단계 3)

영역: 전략

이름: DD_RangeBreak(v 0.1)

Input: len(0.3)

Var1= highd(1)-lowd(1) ‘전일 Range를 var1에 입력

If ttime<1500 then ‘시간이 1500 분 보다 적으면

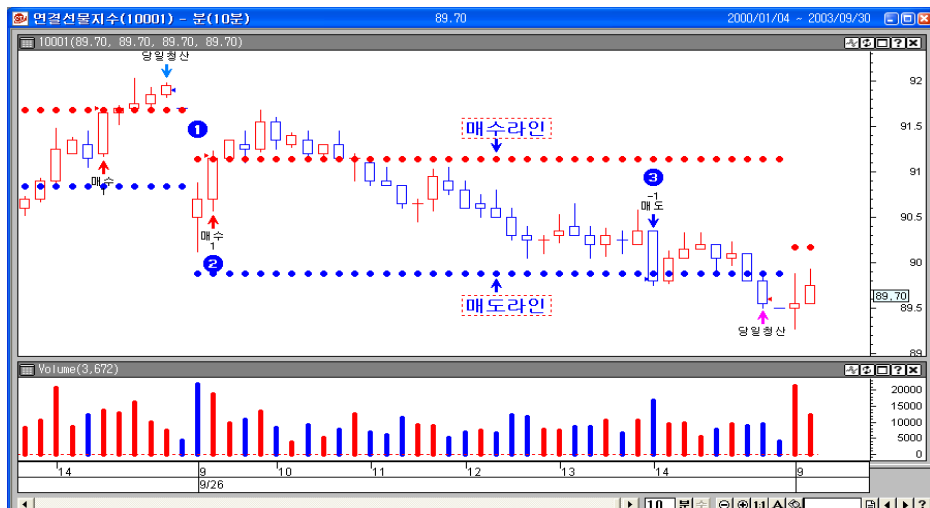
Call buy("매수", Atstop, Def, opend+var1*len) ‘금일 시가 + range*len 돌파시 매수

Call sell("매도", Atstop, Def, opend-var1*len) ‘금일 시가-range*len 돌파시 매도

End if

<수식 3-10>에서는 시간 제어 구문이 주문 명령을 제어하게끔 구성을 하였다. 즉 15:00분 이후에는 주문이 발생하지 않게 하여 그 다음날에 연속적으로 영향을 주지 않게 작성한 것이다. 여기서 15:00은 조절이 가능하다. 예를 들어 14:30 분 이후에는 신규 진입을 금지하기 위해서는 “ttime < 1430”으로 바뀌주면 된다. 결과를 보면 다음과 같다.

<그림 0-12> 시간 제어 추가



①에 발생하였던 잘못된 진입이 제어되고, ②③의 올바른 신호만이 발생하는 것을 알 수 있다. 추가적으로 데이트레이딩 전략을 작성할 경우, 고려해야 할 부분은 “당일 진입금지” 이다.

〈그림 0-13〉 속성창에서 당일 진입금지 설정

선물일 경우 15:05분부터 15:15분까지는 동시호가 시간이다. 만약 이 시간대에 진입이나 청산이 발생한다면 실제매매에서는 불가능한 주문이 된다. 특히 atmarket 주문일 경우 10분 차트에서 15:10분 봉에서 진입 조건이 완성되어 15:15 분 봉에 주문을 하라는 신호가 발생한다면, 실제로 동시호가에 시장가 주문을 내야 하는 문제가 발생한다. 따라서 “당일 진입금지”를 체크하고 그 시간대를 “당일 청산” 시간대 이후로 설정해 놓고 전략을 적용해야 한다.

매매 포인트에 오류가 없는 것이 확인되었다. 그러면 수익은 어떤지 확인해보자.

<표 3-6>과 <표 3-7>을 비교해보면 코딩 오류 및 시간 제어를 거친 결과가 실제 원하던 전략식이며, 그 결과 상당히 좋아진 것을 알 수 있다. 특히 보상비율이 4.41에서 10.92로 위험대비 수익구조가 상당히 안정적으로 발전하였다.

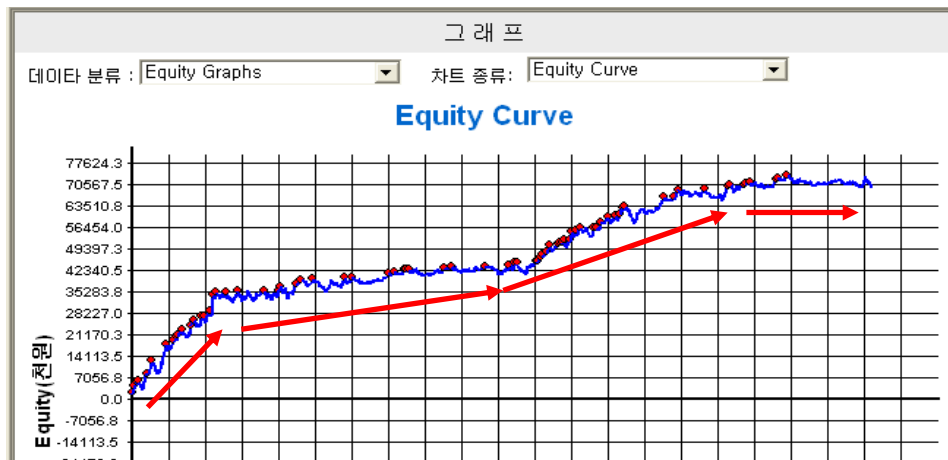
〈표 0-7〉 테스트 결과

평가 항목	전 체	매 수	매 도
총 손익	71,704.88	32,500.25	39,204.63
평균 손익	64.54	58.04	71.15
최대 손실	-1,731.31	-1,731.31	-1,529.01
총 매매수	1,111	560	551
연속 이익 매매수	10	7	9
연속 손실 매매수	6	6	8
최대 자본 인하액	-6,567.95	-5,317.70	-6,899.01
승률	51.94	53.04	50.82
손익비	1.31	1.29	1.34
평균손익비	1.22	1.14	1.30
보상 비율	10.92	6.11	5.68

주) 단위: 천원, 테스트기간: 2000. 01. 04 ~ 2003. 09. 30, 필수적으로 보유해야 할 바 수: 50, 슬리피지: 1틱

누적 수익곡선을 보면 다음과 같다.

〈그림 0-14〉 누적 수익곡선



〈그림 3-14〉를 보면 〈그림 3-10〉와 비교할 때, 수익곡선의 방향성이 좀더 좋아진 것을 알 수 있으며, 수익이 안 나는 구간인 〈그림 3-5〉의 ②④구간에서 횡보 내지 소폭 상승으로 전환 되었음을 알 수 있다. 또한 2001년, 2003년에

서 시작하더라도 큰 손실이 없을 것이라는 것을 예상할 수 있다.

처음 과정인 진입 전략과 진입 전략의 오류 체크가 진행되었다. 이제 다음으로 손절매(stoploss)를 적용해보자.

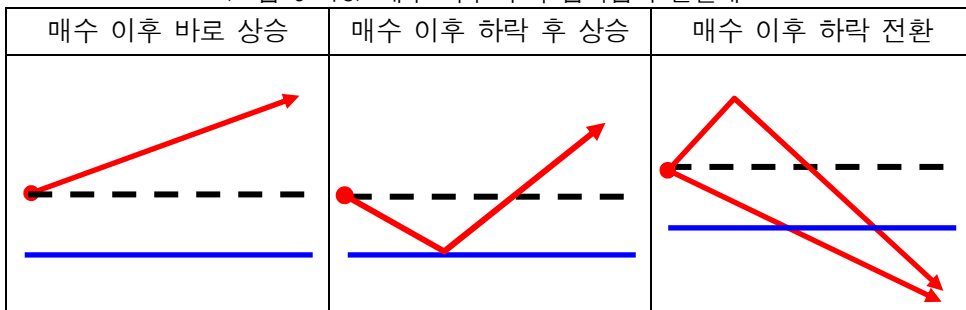
(3) 손절매 적용

손절매는 손실을 최소화하여 수익을 증가시키는 목적으로 사용된다. 어떤 시스템이든 예상치 못한 급변이 발생할 수 있다. 따라서 매수, 매도가 모두 포함된 전략식이라도 손절매(stoploss)를 적용해 둘 필요가 있다.

또한 손절매는 매매에 대한 최소 방어선으로 매수 손절매가 발생한다면, 매수 방향으로 더 이상 진행하지 않을 것이라는 의미를 가지고 있다. 따라서 진입 시점과 가까운 손절매를 설정할 경우 잦은 손절매로 수익이 감소하게 되며, 넓은 손절매의 경우에는 한번의 손실이 커지는 효과가 있어 적절한 손절매를 설정하는 것이 중요해진다.

일반적으로 실전투자자들은 좁은 손절매를 선호한다. 왜냐하면 잦은 매매를 하더라도 빠른 손절매가 심리적 안정을 주기 때문이다. 하지만 위험을 감소시키고, 수익을 키우는 손절매는 가격이 적절히 움직일 수 있게 충분한 공간을 확보한 다소 넓은 손절매가 바람직 하다. 적절한 손절매 폭을 설정하는 방법을 그림으로 설명하면 아래 그림과 같다.

〈그림 0-15〉 매수 이후 주가 움직임과 손절매



<그림 3-15>는 매수 이후 예상 주가 움직임이다. 매수 이후 바로 상승하는 경우, 잠시 하락하였다가 재상승하는 경우와 매수 이후 하락 전환하는 경우인 3가지로 구분할 수 있다. 적절한 손절매 폭은 매수 이후 하락 후 상승하는 경우에는 하락폭보다 넓게 설정되어야 하며, 매수 이후 하락 전환하는 경우에는 손실을 최소화하기 위해 좁게 설정해야 하므로 이 두가지를 동시에 만족하는 가격대를 손절매 폭으로 설정해야 한다.

적절한 손절매 폭을 찾는 방법은 두가지이다. 첫째는 최적화(optimization)로 설정하는 방법과 Maximum Adverse Excursion(MAE)을 이용하는 방법이다. MAE를 이용하는 방법은 5장 청산전략에서 자세히 설명하기로 하고, 여기서는 최적화를 통한 손절매 설정법을 사용하기로 하자.

손절매를 설정하는 방법은 다음과 같다.

- 절대금액이나 포인트 방법(예: 500,000 원, 1 포인트)
- 비율(%)를 사용하는 방법(예: 진입가의 1%)
- 변동성을 이용해서 가변적인 손절매를 설정하는 방법(예: ATR, Range이용)

절대금액이나 포인트를 이용하는 방법이 일반적일 수 있지만, 선물가격 수준이 늘 같지 않다는 관점에서 비율을 사용하는 것이 바람직하다. 더 나아가 시장의 흐름에 따라 변화되는 가변적인 방법도 권장할 만한 방법이 된다. 자세한 가변적인 방법에 대해서는 청산전략 부분에서 다루기로 하고, 여기서는 비율을 사용하는 방법을 먼저 사용하기로 한다.

1) 비율을 이용한 손절매

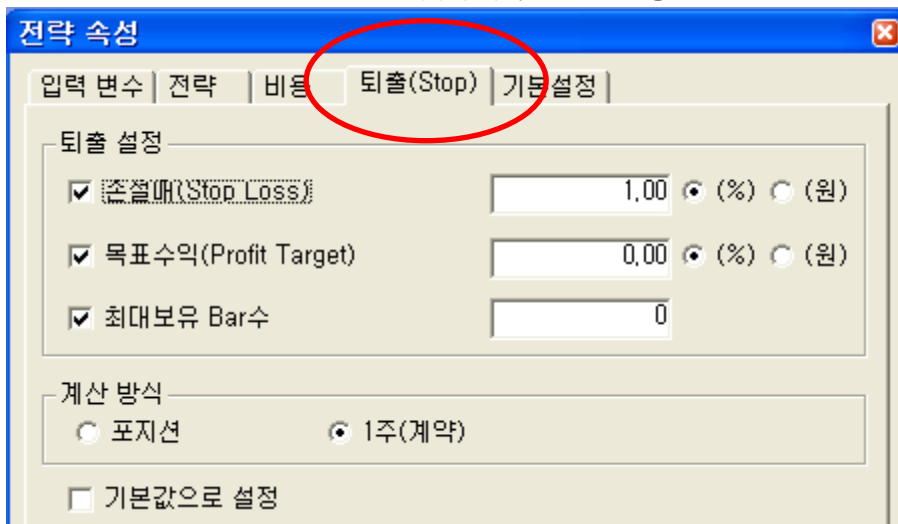
비율을 적용하는 방법은 두 가지 이다. 속성창에서 적용하는 방법과 코딩으로 적용하는 방법이다. 우선 속성창에서 적용하는 방법은 다음과 같다.

속성창에서 퇴출 항목에는 손절매, 목표수익, 최대 보유 바 수가 있는데, 이중 손절매를 선택하고 1 %를 선택한 후 계산 방식에서 계약수나 포지션을 선

택하면, 손절매가 적용되게 된다. 목표 수익은 일정 수준(예 2%)에 도달하면 바로 청산하는 이익보존 청산 전략이고, 최대 보유 바수는 기간 청산으로서, 예를 들어 10이라면 매수 이후 10봉이 지나면 무조건 청산하는 방법이다.

다양한 청산 전략을 사용하기 위해서는 코딩을 통하는 방법이 바람직하다. 다만 코딩을 하는 경우와 속성창에서 선택하는 방법은 차이가 있는데, 이 차이는 꼭 기억하기 바란다. 그 차이는 진입 봉에서 손절매가 작동하느냐이다. 즉, 속성창에서 손절매를 적용하면 해당 진입 봉에서 손절매 조건에 맞으면 손절매가 작동하게 된다. 하지만 코딩으로 손절매를 설정할 경우 진입 봉에서는 손절매가 작동하지 않고, 그 다음 봉에서부터 적용된다.

〈그림 0-16〉 속성창에서 손절매 적용



그러면 이제부터 코딩을 통해서 손절매를 적용하고 최적의 손절매 포인트를 찾는 과정을 시작해보자.

<수식 0-11> 데이트레이딩 전략(단계 4)

영역: 전략

이름: DD_RangeBreak(v 0.1)

Input: len(0.3), len1(0.01)

```
Var1= highd(1)-lowd(1)   '전일 Range를 var1에 입력
If ttime<1500 then   '시간이 15시00분보다 적으면
Call buy("매수", Atstop, Def, opend+var1*len)   '금일 시가+ range*len 돌파시 매수
Call sell("매도", Atstop, Def, opend-var1*len)   '금일 시가-range*len 돌파시 매도
End if
If position <0 then   '현재 포지션이 없지 않다면, 1=매수 포지션, -1 = 매도 포지션
    Call exitlong( "매수손절" , Atstop, Entryprice*(1-len1))   '진입가 대비 매수 손절
    Call exitshort( "매도손절" , Atstop, Entryprice*(1+len1))   '진입가 대비 매도 손절
End if
```

위와 같이 전략 식에다 손절매를 추가하여 재 적용하면, 손절매가 작동하게 된다. 최적화를 하기 전에 오류 검증부터 해보자

(4) 코딩 오류 및 최적화

손절매나 이익보존을 적용하는 과정에서 atmarket, onclose 주문의 경우에는 별 오류 문제가 없지만, Atstop, Atlimit의 경우에는 추가 진입이 바로 발생하는 심각한 오류가 있을 수 있으므로 오류 체크가 아주 중요해진다. 현재 손절매로 Atstop 주문을 사용하였는데, 이런 경우에는 진입가보다 낮은 가격에서 청산이 발생하기 때문에 코딩 오류 문제는 발생하지 않는다.

우선 신호로 손절매 포인트를 표시해보자.

<수식 0-12> 데이트레이딩 전략(단계 5)

영역: 신호

이름: DD_RangeBreak(v 0.1) (DD: 데이트레이딩, 버전은 0.1 이라는 의미)

Input: len(0.3), len1(0.01)

Var1=highd(1)-lowd(1)

Call plots1("매수라인", open+var1*len)

Call plots2("매도라인", open-var1*len)

If i_position =1 Then '매수 포지션,

Call plots3("매수손절", i_Entryprice*(1-len1)) '진입가 대비 매수 손절표시

End If

If i_position=-1 Then '-1 = 매도 포지션

call plots3("매도손절", i_Entryprice*(1+len1)) '진입가 대비 매도 손절 표시

End If

<수식 3-12>는 신호 영역으로 진입과 청산 시점을 표시하는 것인데, 자세히 살펴보면 <수식 3-10>과는 다른 i_position과 i_Entryprice과 같은 함수가 사용된다. i_ 계열 함수로서 사용자 함수, 신호, 지표, 전략 등 모든 영역에서 사용할 수 있는 함수이다. 하지만 “i_” 가 없는 전략 함수는 전략 영역에서만 사용하여야 한다(“알기 쉬운 사이보스트레이더 언어 활용법” 책자를 참조).

<그림 0-17> 손절매 적용화면



<그림 3-17>을 보면 손절매가 잘 적용되는 것을 알 수 있다. 그러면 어느 정도 손절매 포인트가 적절일까? 진입 변수(0.3)와 함께 테스트를 해볼 수도 있지만, 여기서는 손절매 포인트만 최적화를 통해서 알아보기로 하자.

우선 최적화의 범위를 생각해보자. 0.01(1%)일 경우 100 포인트라면 1 포인트가 되며 0.05 라면 5포인트가 된다. 5포인트라면 데이트레이딩에서는 무척 큰 포인트이다. 따라서 0.005(0.5%=0.5 포인트)에서 0.03 까지 0.005 씩 증가시키면서 최적화를 보기로 하자.

<그림 0-18> 최적화 변수 입력

<표 0-8> 최적화 결과

변수	총손익	평균 손익	건수	승률	손익비	평균 손익비	보상 비율	최대 자본인하액
0.005	48,254.39	29.16	1,655	34.56	1.19	2.26	7.41	-6,515.44
0.01	57,434.08	46.36	1,239	46.49	1.24	1.42	8.6	-6,677.32
0.015	63,640.12	55.78	1,141	50.57	1.27	1.24	9.22	-6,901.53
0.02	69,844.17	62.42	1,119	51.56	1.30	1.22	10.63	-6,567.95
0.025	71,577.64	64.37	1,112	51.89	1.31	1.22	10.90	-6,567.95
0.03	71,247.75	64.13	1,111	51.94	1.31	1.21	10.85	-6,567.95

주) : 항목별 최적변수값, 단위: 원, 배, %

최적화 결과 모든 변수에서 유의적인 결과가 도출되지만, 0.01 이하에서는 수익이 급격히 줄고 매매 건수가 급격히 늘어나는 문제가 발생한다. 즉 100 포인트 기준 1포인트 이하의 손절매는 너무 가까운 손절매 포인트라는 것을 알 수 있다.

또 하나 특징은 매수와 매도가 동시에 작동하고 있기 때문에 0.01 이상에서는

손절매 포인트의 변화에 그리 민감하게 반응하지 않음을 알 수 있다. 이 부분은 전략에 따라 차이를 보일 것이다. 이 전략처럼 매수, 매도가 동시에 작동하도록 구성된 전략일 경우 매수 시점에서는 매도선이, 매도시점에서는 매수선이 손절매 역할을 하게 되는 것이다.

또한 <표 3-8>결과에 보듯이 최적 변수인 0.025를 사용하더라도, 손절매를 적용하지 않은 결과값인 71,704,880원에 미치지 못한다는 것이다. 그러면 손절매를 적용해야 하는가? 에 의문이 생긴다.

손절매의 효과는 한번 매매에서 발생하는 최대 손실이 제어되기 때문에 사용되는 것이다. 0.025 와 0.02 의 위험과 수익 구조가 그리 차이가 없다(<표 3-9>). 편의상 100포인트 기준 2포인트 결과를 보기로 하자.

<표 0-9> 테스트 결과

평가 항목	전체	매수	매도
총 손익	69,844.17	32,722.90	37,121.27
평균 손익	62.42	58.12	66.76
최대 손실	-1,215.46	-1,215.46	-1,213.25
총 매매수	1,119	563	556
연속 이익 매매수	10	7	9
연속 손실 매매수	6	6	8
최대 자본 인하액	-6,567.95	-5,751.06	-6,889.05
승률	51.56	52.75	50.36
손익비	1.30	1.29	1.31
평균손익비	1.22	1.16	1.30
보상 비율	10.63	5.69	5.39

주) 단위: 천원, 테스트기간: 2000. 01. 04~2003. 09. 30,
필수적으로 보유해야 할 바 수: 50, 슬리피지: 1틱

<표 3-9> 과 <표 3-7>을 비교해 보았을 때 총손익은 소폭 줄었지만 한번 매매에서 발생할 수 있는 최대 손실에 주목할 필요가 있다. <표 3-7>에서는 최대 손실이 -1,731,310원이었는데 여기서는 -1,215,460원(약2.4 포인트)으로 줄었다. 즉 한번 매매에서의 손실 한도가 약 2 %로 고정되는 것이다.

이와 같이 손절매를 적용함으로 한번 매매에서 발생하는 손실 폭을 제한하게 되어 최대 손실을 관리하게 된다.

전략에 따라 손절매 폭은 다르다. 진입이 전환 포인트(turning point)에 가깝다면 손절매 폭은 그리 크지 않게 설정해도 무방할 것이다. 따라서 손절매 폭의 크기에 따라 전략의 유용성을 파악할 수 있다.

손절매 폭은 앞으로 자산관리와 밀접한 관계를 가지게 된다. 손절매 폭은 한번 매매에서의 예상 최대 손실폭으로서, 이 금액이나 비율을 이용해서 계약수를 결정하게 되는 것이다.

우선 여기서는 손절매를 2~2.5 % 수준이 합리적이라는 결과만 기억하고, 계속 이익 보존청산으로 진행하자.

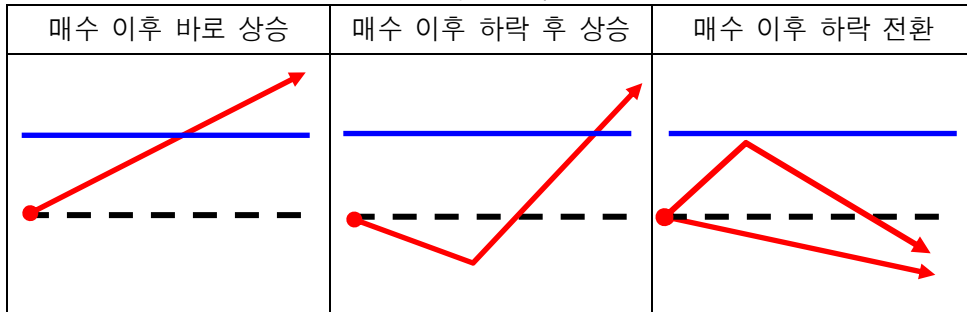
(5) 이익보존 청산 전략

이익 보존 청산 전략은 이익을 극대화하는 청산 전략이다. 손절매와 달리 다양한 전략 유형이 있으며, 매수에서 바로 매도로 전환된다든지 할 경우에는 사용하지 않을 수 있다.

하지만 수익이 발생할 경우 수익을 빨리 실현해서 수익을 극대화하는 방법은 상당한 심리적인 안정을 가져다 준다. 많은 유형의 이익 보존 청산 전략들이 있다. 제 5 장에서 자세히 설명할 것이다.

우선 논리적인 접근부터 하자. 앞서 설명한 손절매와 마찬가지로 진입 이후 일정 수준이 지나고 나면 더 이상 손실을 생각할 필요가 없어지게 되는데, 이 포인트를 손익 분기점(breakeven point)이라 할 수 있다. 매수의 경우를 예로 설명하면 다음과 같다.

<그림 0-19> 매수 이후 추가 움직임과 손절매



<그림 3-19>의 세 번째 그림인 매수 이후 하락 전환 그림을 보면 매수 이후 일정 수준(실선)까지 상승하다 하락으로 전환되는 경우의 점선이 바로 손익 분기점이다.

즉 매수 이후 일정 수준을 넘어가면 더 이상 하락하지 않지만, 일정 수준을 넘지 못하면 손실로 전환 될 가능성이 있다는 것이다. 따라서 적절한 손익 분기점을 설정하고, 그 다음부터는 손절매 대신에 진입가격 이상으로 이익보존 청산 전략을 사용하게 된다.(청산전략 편 참조)

이렇게 이익을 보존하는 형태의 이익보존 청산 전략에는 대표적으로 추적청산(trailing stop)이 있다.

1) 추적청산(trailing stop)

추적청산(추적스탑)도 다양한 종류가 있지만, 원리는 최고 수익 대비 하락폭을 설정하는 방법을 말한다. 앞서 설명한 손절매가 진입가격 대비 최대 하락폭이라고 하면, 이익보존이 가미된 청산 방법은 최고 수익대비 최대 하락폭을 이용하는 것이다.

앞서 설명한 손절매와 추적청산을 동시에 사용하는 등 다양한 추적청산 방법들이 있지만, 여기서는 손절매 대신에 추적청산을 적용하는 방법을 알아보자.

<수식 0-13> 데이트레이딩 전략(단계 6)

영역: 전략

이름: DD_RangeBreak(v 0.1)

Input: len(0.3), len1(0.02)

```
Var1= highd(1)-lowd(1)   '전일 Range를 var1에 입력
If ttime<1500 then      '시간이 15시00분 보다 적으면
Call buy("매수", Atstop, Def, opend+Var1*len)   '금일 시가 + range*len 돌파시 매수
Call sell("매도", Atstop, Def, opend-Var1*len)   '금일 시가-range*len 돌파시 매도
End If

If position <>0 Then '현재 포지션이 없지 않다면, 1=매수 포지션, -1 = 매도 포지션
Call exitlong("추적스탑", Atstop, hhv(1,high, barnumsinceEntry+1)*(1-len1))
'매수 이후 최고 고가 대비 일정 비율 하락하면 청산하라
Call exitshort("추적스탑", Atstop, llv(1,low, barnumsinceEntry+1)*(1+len1))
'매도 이후 최고 저가 대비 일정 비율 상승하면 청산 하라
End If
```

<수식 0-14> 데이트레이딩 전략(단계 7)

영역: 신호

이름: DD_RangeBreak(v 0.1)

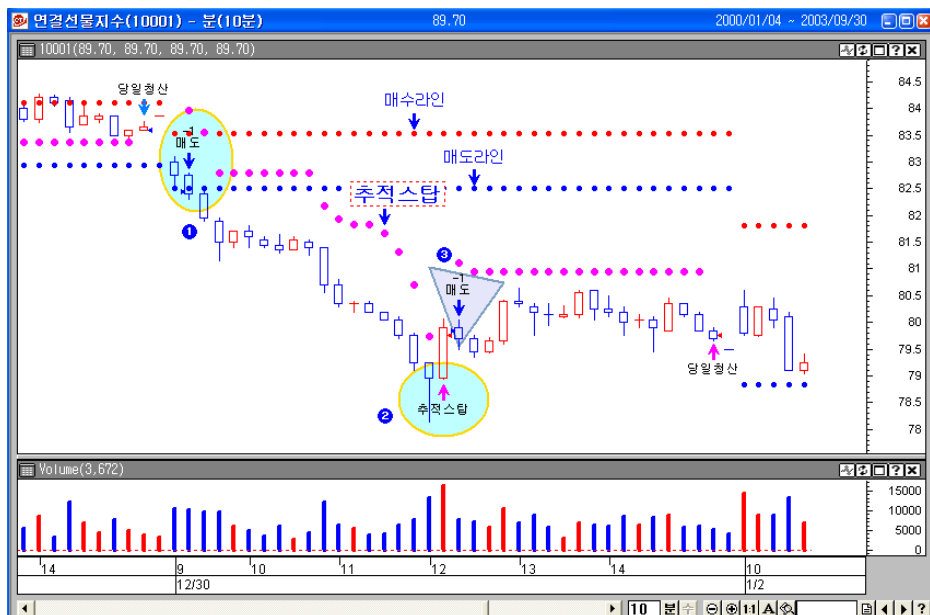
Input: len(0.3), len1(0.02)

```
Var1=highd(1)-lowd(1)
Call plots1("매수라인", opend+Var1*len)
Call plots2("매도라인", opend-Var1*len)
If i_position=1 Then '1=매수 포지션
Call plots3("추적스탑", hhv(1,high,i_barnumsinceEntry+1)*(1-len1))
'매수 이후 최고가 대비 2%하락시 청산
End If
If i_position=-1 Then '-1=매도 포지션
Call plots3("추적스탑", llv(1,low,i_barnumsinceEntry+1)*(1+len1))
'매도 이후 최저가 대비 2% 상승시 청산
End If
```

<수식 3-13>은 손절매 식 대신 매수의 경우, 진입 이후 최고 고가(= 최고수익)에서 일정비율(예 2%) 하락하면 청산하라는 추적스탑을 적용한 것이다. 이렇게 되면 추적청산이 손절매 역할까지 동시에 수행하게 된다. <수식 3-14>는 전략에 대한 신호식이다.

차트를 보면서 추적스탑을 이해해보자

<그림 0-20> 추적스탑



매도 라인을 하향돌파하는 시점에서 ① 번 매도가 발생하고, 추적스탑이 최고 수익대비 일정비율(0.02)을 유지하면서 진행하는 것을 알 수 있다. 결국 ② 번 시점에서 추적스탑이 작동해서 수익을 보존하게 된다.

그런데 ③번 매도가 다시 발생한다. 왜 일까? Atmarket의 경우와 달리 Atstop 매도의 경우 일정 수준 이하(여기서는 시가-전일 $\text{range} \times 0.3$)가 되면 무조건 매도가 발생한다(매수도 동일). 따라서 추적스탑으로 포지션이 없어진 상태에서 매도 조건을 만족하기 때문에, 다시 매도가 바로 발생하게 되는 것이다. 이러한 오류는 이익보존스탑을 사용할 경우 자주 발생하는 오류로서 코딩 수정을 반드시 해야 한다.

(6) 오류 수정

<수식 0-15> 데이트레이딩 전략(단계 8)

영역: 전략

이름: DD_RangeBreak(v 0.1)

Input: len(0.3), len1(0.02)

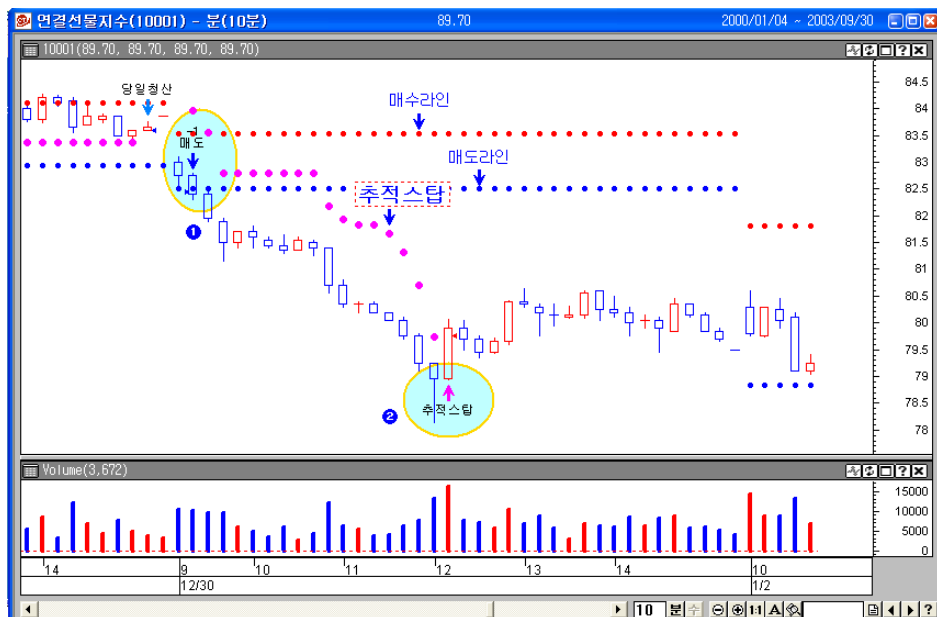
```
Var1= highd(1)-lowd(1)   '전일 Range를 var1에 입력
Cond1= exitname(1)="매수추적스탑" And tdate=exitdate(1) _
And exitprice(1)>opend+Var1*len
    ' 청산이름이 "매수추적스탑" 이고 청산일이 오늘이며 청산가격이 매수 진입가격보다 크면
    ' true 아니면 false
Cond2= exitname(1)="매도추적스탑" And tdate=exitdate(1) _
And exitprice(1)<opend-Var1*len
    ' 청산이름이 "매도추적스탑" 이고 청산일이 오늘이며 청산가격이 매도 진입가격보다 적으면
    ' true 아니면 false
If ttime<1500 Then
    If Cond1=False Then   ' cond1 이 거짓이면
        Call buy("매수", Atstop, Def, opend+Var1*len)
    End If
    If Cond2=False Then   ' cond2가 거짓이면
        Call sell("매도", Atstop, Def, opend-Var1*len)
    End If
End If
If position <>0 Then '현재 포지션이 없지 않다면, 1=매수 포지션, -1 = 매도 포지션
Call exitlong("매수추적스탑", Atstop, hhv(1, high, barnumsinceEntry+1)*(1-len1))
'매수 이후 최고 고가 대비 일정 비율 하락하면 청산하라
Call exitshort("매도추적스탑", Atstop, llv(1, low, barnumsinceEntry+1)*(1+len1))
'매도 이후 최고 저가 대비 일정 비율 상승하면 청산 하라
End If
```

위와 같은 오류를 수정하기 위해서는 진입 제어를 해야 한다. 우선 청산 시기 및 가격 등으로 진입을 제어하는 방법을 알아보자.

<수식 3-15>에서 두 가지 condition문장(cond1,2)으로 진입 부분을 제어하였다. 우선 cond1 은 매수를 제어하는 부분으로서 exitname(1)="추적스탑" 은 직전 청산 이름이 “추적스탑”이면 cond1=true 가 되는 부분이다. 그런데 청산이 오늘 발생한 청산에만 해당하므로 tdate=exitdate(1) 청산일이 오늘이어야 cond1=true 가 된다. 그 다음으로 매수진입가격 보다 낮은 청산 가격은 손절매로서 우리가 제어하려는 부분은 이익 발생 이후에 재 진입이므로 $exitprice(1) > open + var1 * len$ 으로서 이익이 발생할 경우에만 cond1= true 가 된다. 즉 추적스탑이고 오늘 청산이 발생하였고, 이익이 발생한 이익보존 청산 전략일 경우는 cond1= true가 된다.

그러면 매수 진입 부분에서는 cond1=false 인 경우에만 매수 진입을 하면 된다. 매도 부분도 마찬가지이다. 신호 선을 수정할 필요가 없으므로 이제 차트를 보면서 제어가 잘 됐는지 확인을 해보자.

<그림 0-21> 추적스탑(오류 제어)



앞서 본<그림 3-21>의 ③번 매도가 제어되었다. <수식 3-15>로 손절매 + 이익보존 추적스탑이 적용된 전략식이 완성되었다. 식에 더 이상 오류가 없는 상태이다. 그러면 테스트 결과를 보자.

〈표 0-10〉 테스트 결과

평가 항목	전체	매수	매도
총 손익	75,033.25	32,404.29	42,628.95
평균 손익	65.19	56.16	74.27
최대 손실	-1,215.46	-1,215.46	-1,213.25
총 매매수	1,151	577	574
연속 이익 매매수	10	9	9
연속 손실 매매수	6	6	7
최대 자본 인하액	-6,145.95	-6,949.57	-5,435.05
승률	51.17	51.99	50.35
손익비	1.34	1.29	1.39
평균손익비	1.28	1.19	1.37
보상 비율	12.21	4.66	7.84

주) 단위: 천원, 테스트기간: 2000. 01. 04~2003. 09. 30,
필수적으로 보유해야 할 바 수: 50, 슬리피지: 1틱

<표 3-9>와 비교 시, 수익이 증가하고 위험은 감소하여 보상비율이 10.63에서 12.21로 높아진 것을 알 수 있다. 비교 분석을 위해서 3가지 결과 비교 분석해보자.

<표 3-11>을 보면 손절매와 추적스탑이 적용된 전략의 매매수가 증가하면서 수익이 증가하였고 위험(최대손실, 최대자본인하액)이 감소하였으며 손익비, 평균손익비, 보상 비율이 증가하였다.

즉 수익은 증가하고 위험은 감소한 형태로 전략의 유용성이 증가한 것을 알 수 있다.

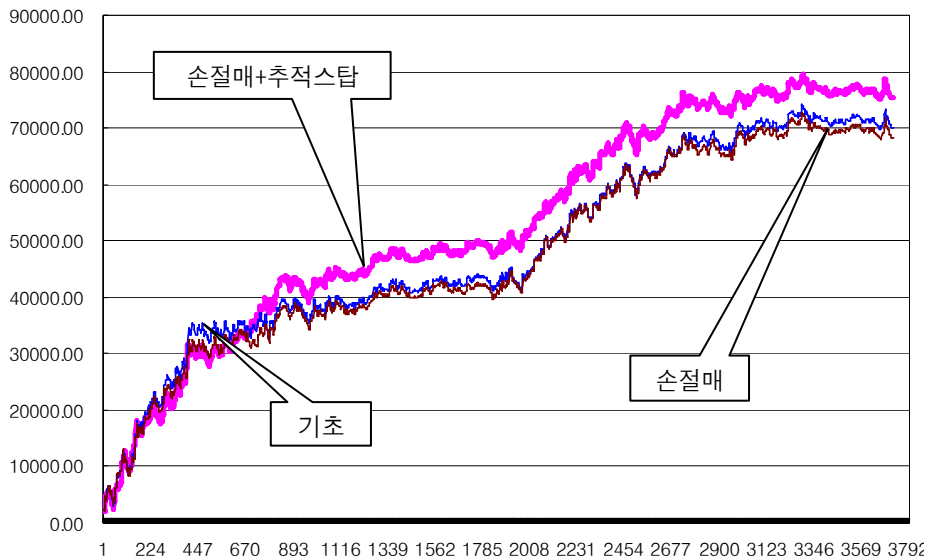
〈표 0-11〉 테스트 결과 비교 분석(전체)

평가 항목	기초	손절매	손절매+추적스탑
총 손익	71,704.88	69,844.17	75,033.25
평균 손익	64.54	62.42	65.19
최대 손실	-1,731.31	-1,215.46	-1,215.46
총 매매수	1,111	1,119	1,151
연속 이익 매매수	10	10	10
연속 손실 매매수	6	6	6
최대 자본 인하액	-6,567.95	-6,567.95	-6,145.95
승률	51.94	51.56	51.17
손익비	1.31	1.30	1.34
평균손익비	1.22	1.22	1.28
보상 비율	10.92	10.63	12.21

주) 단위: 천원, 테스트기간: 2000. 01. 04~2003. 09. 30,
필수적으로 보유해야 할 바 수: 50, 슬리피지: 1틱

누적 수익곡선도 위와 같이 비교 분석해보자

〈그림 0-22〉 누적 수익곡선 비교 분석



<그림 3-22>를 보면 기초와 손절매는 수익곡선이 거의 차이가 없으나, 추적스탑을 적용한 수익곡선은 증가추세가 더 큰 것을 알 수 있다.

이와 같이 “진입 → 오류 수정 → 손절매 → 오류수정 → 이익보존청산 → 오류 수정”과정을 통해서 수익은 증가시키고, 위험은 감소시켜 유용한 전략을 완성하게 된다. 초보자라면 위와 같은 개발 과정을 반복적으로 진행해야 하지만, 조금만 숙달이 되면 상당히 빠른 시간에 개발이 가능하게 된다.

이제 마지막 단계로 자산 관리 방법을 적용해서 실전 매매를 준비하게 된다. 자산관리 또한 수익을 증가시키고, 위험을 감소시키는 방법으로 실제 시장에서 살아남기 위해서는 아주 중요한 부분이다.

아주 중요함에도 불구하고 실제로는 많이 인식을 못하는 편이고, 다소 난해하게 생각되는 부분으로서 “따라하기”에서는 다루기가 힘든 것으로 판단된다. 따라서 6장 자산관리에서 자세히 다루기로 하자.

여기서는 우선 위 전략이 가지고 있는 문제 등을 좀더 개선하는 방향과, 다양한 제어를 통해서 변형 전략을 만들어 보기로 하자.

(7) 데이트레이딩전략의 개선 및 변형

앞서 설명한 DD_RangeBreak(v0.1) 전략을 좀더 유용한 전략이나, 변형된 전략으로 만들어 보자. 여기서는 진입 조건을 추가한다기 보다, 최적화 및 다양한 진입 제어 등을 이용해서 전략을 개선, 변형시켜 보기로 하자. 다양한 청산 전략을 추가하거나, 변형하는 부분은 5장 청산전략 편에서 해보도록 하자.

앞서 손절매 포인트를 설정하는 과정에서 최적화를 시행하였다. 여기서는 <수식 3-15>의 진입변수 $len(0.3)$ 과 추적스탑 청산 변수 $len1(0.02)$ 를 동시에 최적화하여 최적 변수를 찾아 보도록 하자.

1) 최적화(Optimization)

최적화는 입력 변수와 추적스탑 변수를 동시에 최적화한다.

<표 0-12> 최적화 결과

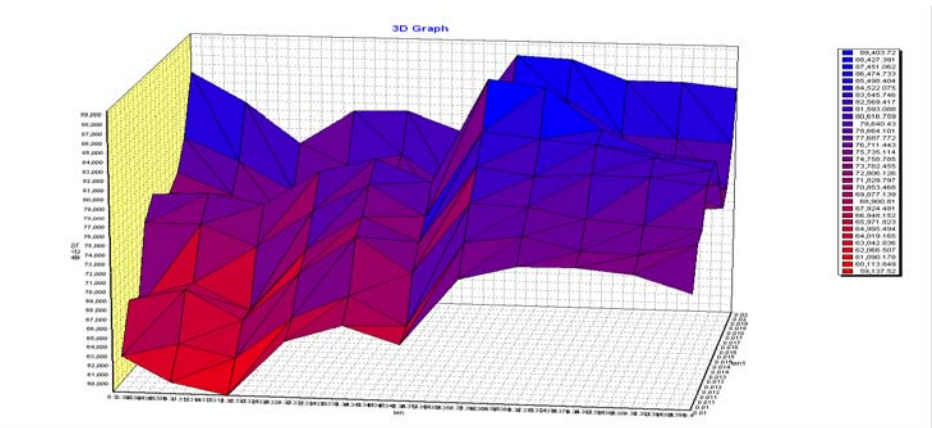
len	Len1	총손익	평균 손익	건수	승률	손익비	평균 손익비	보상 비율	최대 자본 인하액
0.4	0.02	★76,472.4	81.8	934	53.3	1.46	1.28	12.8	-5,968.4
0.6	0.03	44,820.0	71.3	628	55.8	1.40	1.11	7.88	-5,688.4
0.6	0.02	49,193.8	76.0	647	55.4	1.46	1.17	11.0	-4,434.5
0.1	0.03	40,071.4	18.6	2,153	34.6	1.10	2.08	2.5	-15,992.3
0.4	0.015	★72,557.1	74.2	977	52.0	1.45	1.34	13.9	-5,199.9
0.6	0.01	38,487.7	50.6	760	47.8	1.40	1.52	9.8	-3,893.5

주) 진입 변수: 0.1~1, 증가치 0.1, 추적스탑 변수: 0.005~0.03, 증가치 0.005

■ : 각 항목별 최적 변수

<표 3-12>는 각각의 주요 판단 항목에서(건수 제외, 최대자본인하액은 최소 값) 최대값만을 표시하였는데, 진하게 표시된 부분이 해당 항목을 의미한다. (0.4, 0.02)조합의 경우 총손익과 평균손익 기준으로 최적변수이며, (0.4, 0.015)조합은 보상비율기준으로 최적변수임을 알 수 있다. 따라서 최적 변수로 두 조합 다 사용이 가능하다.

<그림 0-23> 3D 차트(총손익 기준)



그림을 보면 넓은 구간에서 수익이 발생하는 광범위 변수인 것을 확인할 수 있다. 따라서 범용적인 전략으로서 어느 변수를 사용해도 무방하다는 것을 알 수 있다.

이제 좀더 세밀한 최적화를 해보자. 우선 진입 변수는 0.3~0.4 가 좋다는 것을 알았다. 그러면 이 사이를 0.01 증가치로 세밀한 분석을 해본다. 추적스탑 변수는 0.01 ~ 0.02 사이가 좋다. 따라서 이 사이를 0.001 증가치로 다시 세밀한 분석을 해본다.

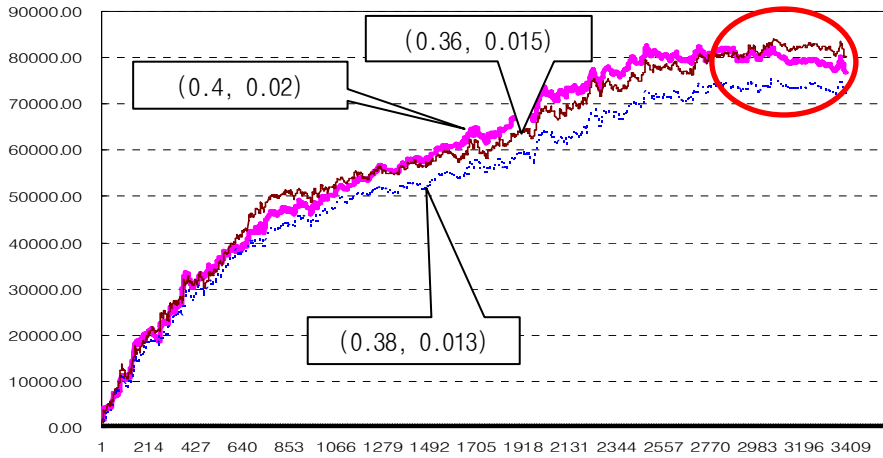
<표 3-13>처럼 총 손익이 다소 많이 차이가 나는(0.36, 0.01)조합을 제외한 나머지 변수들은 상당히 좋은 결과를 나타낸다. (0.4, 0.02)를 가장 유용한 변수로 선정할 수 있으나 (0.36, 0.015)와 (0.38, 0.013)도 최적 변수로 볼 수 있다. 이 세가지 변수에 대한 누적수익곡선을 비교 분석을 해보자.

<표 0-13> 세밀한 최적화 결과(각 항목별 최적 변수)

Len	Len1	총손익	평균 손익	건수	승률	손익비	평균 손익비	보상 비율	최대 자본 인하액
0.36	0.015	80,128.25	75.9	1,055	50.7	1.45	1.41	18.6	-4,295.65
0.4	0.02	76,472.43	81.8	934	53.3	1.46	1.28	12.8	-5,968.48
0.37	0.015	79,540.11	77.1	1,031	51.2	1.46	1.39	17.2	-4,609.64
0.36	0.01	61,076.38	49.8	1,226	44.4	1.35	1.69	13.0	-4,689.01
0.38	0.013	71,884.04	67.8	1,060	50	1.43	1.43	18.8	-3,818.02

주) 입력 변수: 0.3~0.4 증가치 0.01, 추적스탑 변수: 0.01~0.02 증가치 0.001

〈그림 0-24〉 최적 변수 비교(누적수익곡선)



〈그림 3-24〉를 보면 최근 장세에서 (0.4, 0.02)와 (0.36, 0.015)의 수익곡선이 역전되는 것을 알 수 있다. 즉 (0.4, 0.02)의 수익곡선 하락세가 최근 장세에 다소 심하게 발생한다는 사실이다. 따라서 (0.4, 0.02)변수보다는 (0.36, 0.015)변수가 최적 변수가 되며, 최근 장세에서 수익곡선회보를 보이고 있는 (0.38, 0.013)도 사용 가능할 것으로 판단된다.

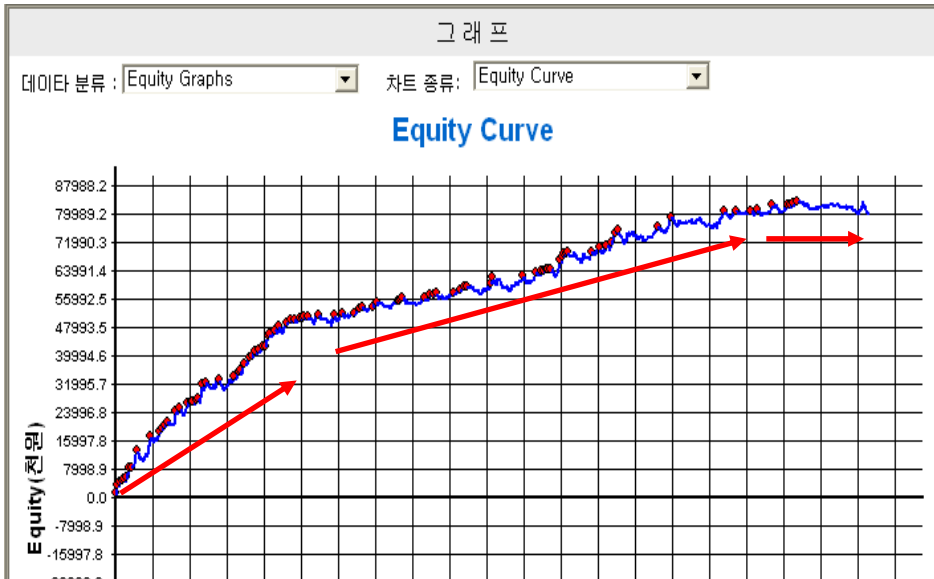
〈표 0-14〉 최적화 결과 (0.36, 0.015)

평가 항목	전체	매수	매도
총 손익	80,128.25	29,314.99	50,813.26
평균 손익	75.95	54.90	97.53
최대 손실	-1,081.98	-1,081.98	-852.71
총 매매수	1,055	534	521
연속 이익 매매수	7	6	8
연속 손실 매매수	6	6	7
최대 자본 인하액	-4,295.65	-5,814.97	-7,253.14
승률	50.71	50.19	51.25
손익비	1.45	1.32	1.60
평균손익비	1.41	1.31	1.52
보상 비율	18.65	5.04	7.01

주) 단위: 천원, 배, %

위험(최대 손실, 최대 자본 인하액) 이 줄어들었고 수익은 증가하여 보상비율이 18.65 로 증가하였음을 알 수 있다.

<그림 0-25> 누적 수익곡선(0.36, 0.015)



<그림 3-25>는 변수 0.36, 0.015의 누적 수익곡선인데 선형성이 개선 되었다는 것을 알 수 있다.

여기서 한가지 짚고 넘어가야 할 문제가 있다. 전략 변수를 테스트 한 결과 대부분의 변수에서 수익이 발생하는 범용적인 전략이다. 그런데 지금의 최적화는 과 최적화(over fitting)의 문제는 없는 것일까?

일단 변수는 2개로서 많은 변수를 사용하지 않았으므로 과 최적화의 범주에 들어가지는 않는다. 하지만 해답은 역시 테스트 이후 전략의 수익곡선이 과거 추이를 유지하는지를 향후 지켜보고 결정해야 할 것이다.

앞서 설명하였듯이 개발이 되었다 해서 바로 실전매매를 해서는 안 된다. 왜냐하면 전략의 추가적인 오류 아니면, 과 최적화 여부를 판단해 볼 수 있도록 가상 매매를 일정 기간 진행해 보아야 한다.

시장은 변화된다. 따라서 변수들도 변화될 가능성 많다. 따라서 긴 기간을 테스트 해서 최적 변수를 찾지만, 이 또한 변화를 겪을 수 있다. 여기에 대한 해결책은 명확하지 않다. 하나의 해결 방법은 가변적인(adaptive) 진입과 청산 전략을 개발하는 것이다.

즉 진입변수값이 전일 변동폭의 0.3이 아니라 시장 상황에 따라 좀더 가변적으로 변화되는 비율을 사용한다든지, 청산에서는 비율이 아니라 변동성(ATR, Range)을 이용한 가변적인 청산 전략을 사용하는 것이다.

진입 부분을 가변적으로 가져가거나, 다른 논리를 적용하는 방법은 4장 진입 전략 부분에서 다루기로 하고, 청산 부분을 가변적으로 변경하는 방법만을 간단히 알아보기로 하자.

2) 가변적인 청산 전략으로 변경

가변적인 청산 전략 등 다양한 청산 전략에 대해서는 5장에서 자세히 설명하겠지만, 여기서는 앞서 사용한 비율 추적스탑을 가변적인 추적스탑으로 변경하여 사용하는 방법에 대해서만 알아보기로 하자.

가변적 추적스탑으로의 변경은 제 3 장 처음 부분에서 설명한 “변동성”을 이용하면 된다. 즉 ATR 이나 Range 등을 사용해서 시장 상황에 맞게 변화되도록 청산 전략을 변경하게 된다.

앞서 설명한 DD_RangeBreak(v0.1)을 그대로 두고 버전 0.2를 새로 작성해서 사용해보기로 하자.

〈수식 0-16〉 가변 청산 전략

영역: 전략

이름: DD_RangeBreak(v 0.2)

Input: len(0.3), atrlen(10), len1(3)

```
Var1= highd(1)-lowd(1)   '전일 Range를 var1에 입력
Cond1= exitname(1)="매수추적스탑" And tdate=exitdate(1) _
And exitprice(1)>opend+var1*len
Cond2= exitname(1)="매도추적스탑" And tdate=exitdate(1) _
And exitprice(1)<opend-var1*len

If ttime<1500 Then
  If Cond1=False Then   ' cond1 이 거짓이면
    Call buy("매수", Atstop, Def, opend+var1*len)
  End If
  If Cond2=False Then   ' cond2가 거짓이면
    Call sell("매도", Atstop, Def, opend-var1*len)
  End If
End If

If position <=0 Then
Call exitlong("매수추적스탑", Atstop, hhv(1, high, barnumsinceEntry+1)- _
atr(atrlen)*len1)
'매수 후 최고 고가 대비 일정기간 동안의 ATR 값의 배수값을 차감한 값을 이탈하면 청산
Call exitshort("매도추적스탑", Atstop, llv(1, low, barnumsinceEntry+1)+ _
atr(atrlen)*len1)
'매도 후 최고 저가 대비 일정기간 동안의 ATR 값의 배수값을 더한 값을 상향하면 청산
End if
```

신호도 동시에 작성해서 차트에 적용하기로 하자.

<수식 0-17> 가변 청산 신호

영역: 신호

이름: DD_RangeBreak(v 0.2)

Input: len(0.36), atrlen(25), len1(3.5)

Var1:=highd(1)-lowd(1)

Call plots1("매수라인", open+var1*len)

Call plots2("매도라인", open-var1*len)

If i_position =1 Then '매수 포지션,

Call plots3("추적스탑", hhv(1, high, i_barnumsinceEntry+1)-atr(atrlen)*len1)

End If

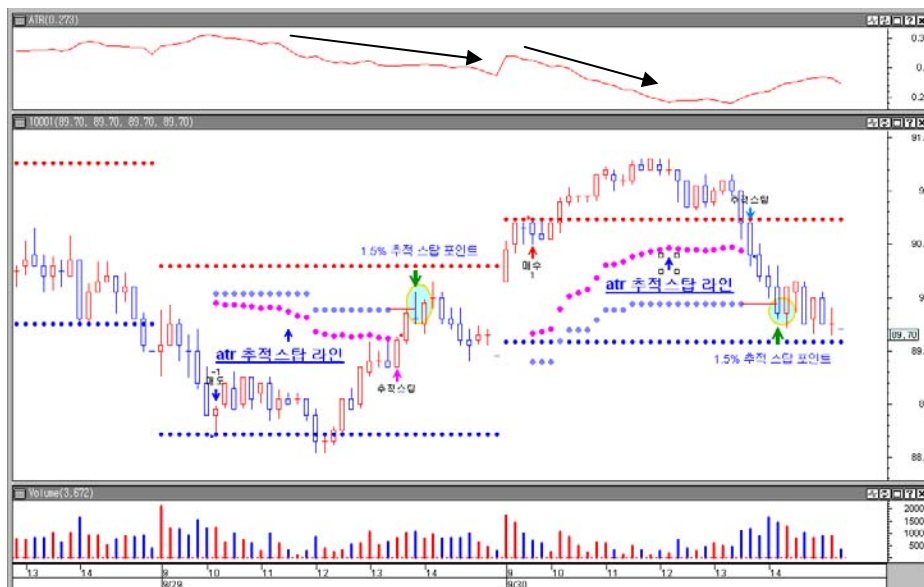
If i_position=-1 Then '-1 = 매도 포지션

Call plots3("추적스탑", llv(1, low, i_barnumsinceEntry+1)+atr(atrlen)*len1)

End If

적용된 그림을 보면서 비율 추적스탑과 ATR 가변 추적스탑을 비교해보자

<그림 0-26> ATR 가변 추적스탑(1)



주) 연결선물 지수 10분 차트: 적용 전략 “DD_RangBreak(v0.2)”

변수: len: 0.36, atrlen: 25, len1: 3.5

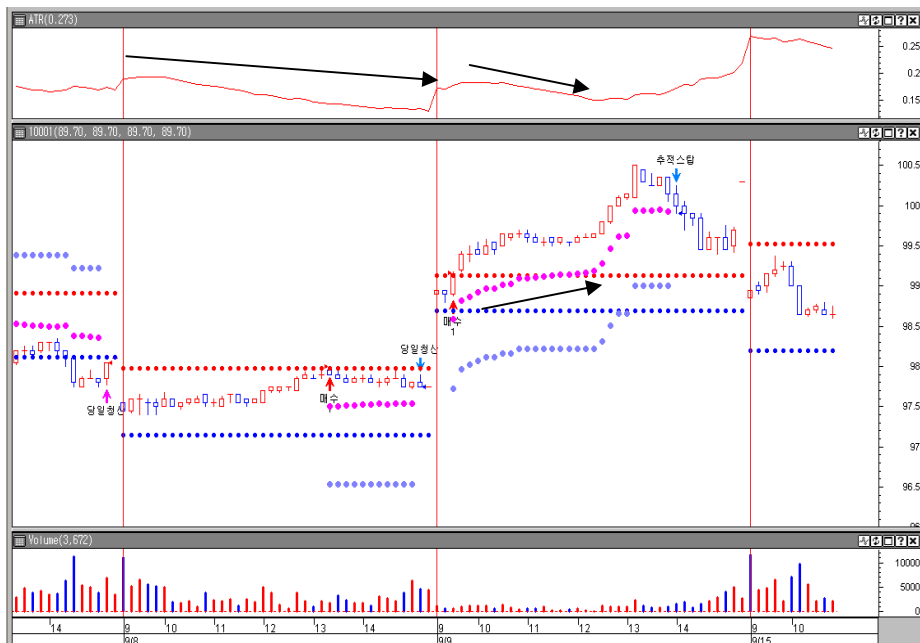
DD_RangeBreakout(v0.1)의 1.5% 추적스탑 포인트 동시 적용

<그림 3-26>에서 ATR 추적스탑이 비율 추적스탑보다 스탱 포인트가 가까운 것을 알 수 있다. 비율 스탱의 경우 지수대가 높은 지수(예: 100)일 경우 상당히 넓은 스탱 포인트를 가지게 된다. 반면 ATR 추적스탑은 스탱 포인트가 현재의 변동성에 따라서 가변적으로 조절된다. 즉 변동성이 적은 시점에서는 진입가격에 가까운 스탱을, 변동성이 큰 시점에서 진입가격에서 넓은 스탱 포인트를 가지게 된다(9월 30일은 전일부터 변동성이 계속 줄어드는 관계로 ATR추적스탑이 진입가격에 상대적으로 가까운 포인트를 설정하게 된다.).

또 하나의 ATR추적스탑의 큰 장점은 진입 이후 가격이 진입과 반대로 움직이거나, 횡보추세가 발달하면 변동성이 감소 하면서(<그림 3-26>의 제일 위쪽 그림은 ATR 지표임) 진입 가격에 근접하는 움직임을 보인다.

반면 비율의 경우에는 최고 수익대비로 고정되어 있어 청산가격대까지 기다려야 하지만, ATR추적스탑은 변동성을 기준으로 하므로 방향성 반대가 되면 청산 포인트가 빠르다는 것을 알 수 있다. 이 부분이 시장의 흐름을 반영하는 ATR 추적스탑의 가장 큰 장점이다.

<그림 0-27> ATR 가변 추적스탑(2)



<그림 3-27>을 보면 9월 8일 장세가 하루 종일 횡보하는 현상을 보였다. 따라서 9월 9일 시점에서 매수가 발생하였는데, ATR가변스탑이 진입가 대비 상당히 가까운 가격대의 추적스탑 포인트를 나타냈다. 상승추세 진행 중에 횡보국면에서는 비율일 경우에 같은 가격대를 가지는 반면, ATR스탑의 경우 최고 수익에 좀더 다가가는 형태를 보이는 것을 알 수 있다.

결론적으로 시장의 변동성에 따라 가변적으로 변화되는 ATR 추적스탑이 실전 매매에서 좀더 유용하게 사용될 수 있음을 알 수 있다. 그러면 최적화 및 테스트 결과를 보자.

최적화는 len, atrlen, len1을 모두 동시에 적용해 보자(1차 넓은 범위 테스트 이후 상세 범위의 경우만 설명).

<표 0-15> ATR 추적스탑 최적화 결과(각 항목별 최적 변수)

len	atr len	len1	총손익	평균 손익	건수	승률	손익비	평균 손익비	보상 비율	최대 자본 인하액
0.37	20	2.7	83,858	73.3	1,143	47.5	1.5	1.66	20.2	-4,143.9
0.4	10	3.8	81,130	83.2	975	51.0	1.51	1.45	13.6	-5,954.7
0.4	30	4	75,571	79.0	956	52.0	1.46	1.34	14.9	-5,047.8
0.4	30	3.3	79,914	79.4	1,006	50.4	1.51	1.48	17.1	-4,664.6
0.36	10	2.1	63,025	47.3	1,330	43.3	1.37	1.79	11.6	-5,416.5
0.38	20	2.7	78,114	69.1	1,129	47.3	1.47	1.64	20.7	-3,765.5

주) len: 3~4 증가치 0.01, atrlen: 10 ~ 30 증가치 10, len1: 2 ~ 4 증가치 0.1

다양한 결과 중 평균손익비를 제외한 나머지 변수들은 유용한 변수들로서, 이중 진입 시점이 빠르거나 중복 발생 변수인 0.37, 20, 2.7 과 0.38, 20, 2.7 을 선택할 수 있을 것이다.

0.37, 20, 2.7를 이용해서 테스트 결과를 비율을 이용한 DD_RangeBreak(v0.1)의 테스트 결과와 비교해보자.

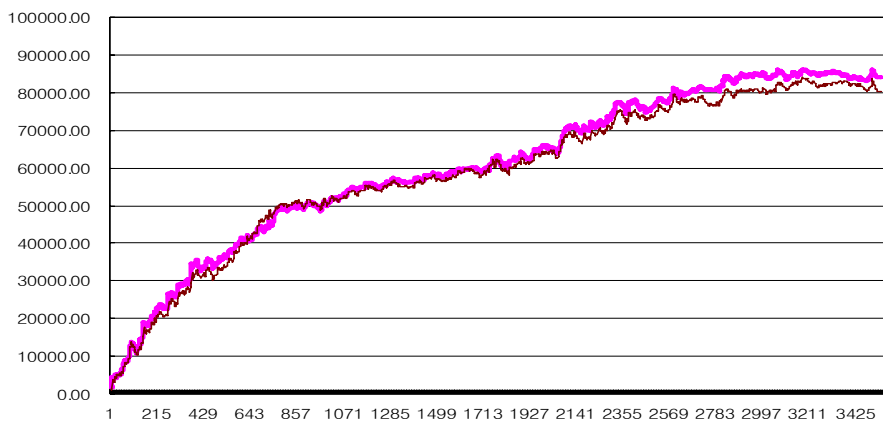
〈표 0-16〉 최적화 결과 비교

평가 항목	Rangebreakout (v0.1)	비교	Rangebreakout (v0.2)
총 손익	80,128.25	<	83,858.50
평균 손익	75.95	>	73.37
최대 손실	-1,081.98	>	-1,132.98
총 매매수	1,055	<	1,143
연속 이익 매매수	7	<	10
연속 손실 매매수	6	<	9
최대 자본 인하액	-4,295.65	<	-4,143.91
승률	50.71	>	47.51
손익비	1.45	<	1.50
평균손익비	1.41	<	1.66
보상 비율	18.65	<	20.24

주) 단위: 천원, %, 배

승률은 감소, 최대 손실은 다소 증가하였지만 DD_RangeBreak(v0.2)가 좀더 향상된 결과를 보임을 알 수 있다. 또한 실제로 비율 1.5% 보다 $\text{atr}(20) \times 2.7$ 배가 전반적으로 진입가격에 가까운 청산가격을 보이고 있어, 실제 매매에서 위험(예상 손실폭)이 적어 좀더 공격적인 전략 운용을 가능하게 해주고 위험 수준이 낮아지는 효과가 있다. 그러면 v0.1 과 v0.2의 수익곡선을 비교해보자.

〈그림 0-28〉 누적 수익곡선(v0.1, v0.2)



<그림 3-28>에서 굵은 실선이 $v = 0.2$, 가는 실선이 $v = 0.1$ 이다. $v = 0.2$ 가 최근 장세에서 수익이 $v = 0.1$ 보다 크다는 것을 알 수 있다. 따라서 앞서 설명한 바와 같이 시장 상황에 가변적으로 적용되며, 비율보다 다소 손절매 폭이 짧다. 횡보 시 빠른 방어 전략이 가능한 가변적인 추적스탑이 좀더 유용하다고 할 수 있다.

DD_RangeBreak($v = 0.2$)을 기준으로 다양한 진입제어를 추가하여 전략을 변경, 개선하는 방법에 대해서 알아보기로 하자.

3) 진입 제어

데이트레이딩에서 진입을 제어하는 이유는 앞에서 본 것과 같이 오류를 수정하는 것 이외에도 다른 이유로 진입을 제어하게 된다.

선물 시장에서 하루에 가격이 추세적으로 움직일 수 있는 시간은 최대 6시간이다. 초 단타매매(아주 짧은 시간대(예 1분, 1틱 등))를 이용하거나, 짧은 목표수익과 손절매를 가져가는 매매와 데이트레이딩을 구분해서 생각한다면, 데이트레이딩일 경우 하루에 1 ~ 5 번 정도의 매매 밖에는 일어나지 않는 게 정상적이다. 왜냐하면 추세가 변화되는 횡수가 적기 때문이다.

아침에 추세가 형성되어 계속 그 추세로 진행되는 경우도 많다. 이런 경우 1 번 진입으로 충분한 수익을 발생하게 된다. 반면 추세가 없는 구간에서는 매매가 반복적으로 일어날 수 있다. 이런 경우에는 가장 쉬운 방법은 매매를 하지 않는 것이 될 것이다.

따라서 과도한 진입이나 손절매 이후 같은 방향 진입을 금지한다든지, 하루에 한번만 매매 한다 등과 같이 진입 횡수나 진입 방향을 제어하는 형태를 자주 사용하게 된다.

그 유형은 아래와 같다.

- 하루에 한번만 매매: 추세가 형성되면 한번 매매로 충분한 수익을 발생할 수 있음
- 당일 매매 횟수 제한: 하루 중 여러 번 매매 발생시 비 추세일로서 수익이 발생할 가능성이 적기 때문임.
- 손절매 발생시 당일매매 중단: 매수 진입에서 손절매가 발생할 경우 오늘은 어떤 방향으로 진입해도 손실이 발생할 것이라는 가정에서 출발함.
- 이익 발생시 추가 진입 제한: 이익이 발생하면 이후로는 수익발생 가능성이 적다는 가정으로 추가 진입을 하지 않음.
- 같은 방향 매매 제어: 매수 이후 청산이 발생한 다음에 다시 매수 방향으로 매매가 발생해도 진입하지 않고 매수 이후 매도나 매수청산 이후 매도만을 진입하는 방법
- 14시 전에 진입이 발생하지 않으면 진입 금지: 강한 장중 추세 형성은 대부분 아침 시간대에 형성될 가능성이 크기 때문에 오후에는 진입 하지 않음.
- 9시 30분 이후 진입: 아침 시간대는 추세를 탐색하는 시간대로 9 시 30분 이후나 10시 이후부터 진입을 허용하는 방법.
- 1차 매매와 2 차 매매의 청산 전략 변경: 하루 중 1차 매매와 2차 이후 매매의 청산 속도를 구분하여 적용하는 형태

이 이외에도 다양한 변형이 가능하다. 중요한 것은 어떤 제어가 좋은 것이냐가 아니라, 자신의 매매 스타일이나 자금 여력 등에 맞게 설정되어야 한다는 점이다.

위의 DD_RangeBreak(v0.2)를 이용해서 위의 진입 제어 중 성과가 좋은 2가지와 이들을 결합한 전략을 구현해 보자.

〈수식 0-18〉 당일 매매 횟수 제한

영역: 전략

이름: DD_RangeBreak(v 0.21)

Input: len(0.37), atrlen(20), len1(2.7), tcount(2)

```

Var1=highd(1)-lowd(1)
If tdate(1)<>tdate Then ‘날이바뀐 첫봉에서
    Var10=currentEntrynum ‘현재까지의 진입횟수를 var10에 저장한다.
End If
Cond1= exittname(1)="매수추적스탑" And tdate=exitdate(1) _
And exitprice(1)>opend+var1*len
Cond2= exittname(1)="매도추적스탑" And tdate=exitdate(1) _
And exitprice(1)<opend-var1*len
If ttime<1500 Then
    If currentEntrynum < Var10+tcount And Cond1=False Then ‘tcount가 20이므로
    ‘당일의 매매가 2회발생하면 조건을 만족하지 못하므로 더 이상 매수하지 못함
        Call buy("매수",Atstop,Def,opend+Var1*len)
    End If
    If currentEntrynum < Var10+tcount And Cond2=False Then ‘tcount가 20이므로
    ‘당일의 매매가 2회 발생하면 조건을 만족하지 못하므로 더 이상 매도 하지 못함
        Call sell("매도",Atstop,Def,opend-Var1*len)
    End If
End If
If position<>0 Then
Call exitlong("매수추적스탑", Atstop, hhv(1, high, barnumsinceEntry+1)- _
atr(atrlen)*len1)
Call exitshort("매도추적스탑", Atstop, llv(1, low, barnumsinceEntry+1)+ _
atr(atrlen)*len1)
End If
    
```

〈수식 0-19〉 같은 방향 진입제어

영역: 전략

이름: DD_RangeBreak(v 0.22)

Input: len(0.37), atrlen(20), len1(2.7)

Var1=highd(1)-lowd(1)

Cond1= tdate=exitdate(1) And position(1)=1 ‘직전의 포지션이 매수이고

‘금일 청산한 적이 있으면 cond1이 true, 아니면 false

Cond2= tdate=exitdate(1) And position(1)=-1 ‘직전의 포지션이 매도이고

‘금일 청산한 적이 있으면 cond2가 true, 아니면 false

If ttime<1500 Then

 If Cond1=False Then ‘직전포지션이 매수가 아니거나, 금일 청산한 적이 없으면,

 Call buy("매수", Atstop, Def, opend+var1*len) ‘매수

 End If

 If Cond2=False Then ‘직전포지션이 매도가 아니거나, 금일 청산한 적이 없으면

 Call sell("매도", Atstop, Def, opend-var1*len) ‘매도

 End If

End If

If position<=0 Then

Call exitlong("매수추적스탑", Atstop, hhv(1,high, barnumsinceEntry+1)- _
atr(atrlen)*len1)

Call exitshort("매도추적스탑", Atstop, llv(1,low, barnumsinceEntry+1)+ _
atr(atrlen)*len1)

End If

<수식 0-20> 매매 횟수 및 같은 방향 진입제어 결합

영역: 전략

이름: DD_RangeBreak(v 0.3)

Input: len(0.37), atrlen(20), len1(2.7), tcount(2)

```

Var1=highd(1)-lowd(1)
If tdate(1)<>tdate Then
    Var10=currentEntrynum
End If
Cond1= tdate=exitdate(1) And position(1)=1
Cond2= tdate=exitdate(1) And position(1)=-1
If ttime<1500 Then
    If currentEntrynum < Var10+tcount And Cond1=False Then '<수식 3-18, 19> 참조
        Call buy("매수", Atstop, Def, opend+var1*len)
    End If
    If currentEntrynum < Var10+tcount And Cond2=False Then
        Call sell("매도", Atstop, Def, opend-var1*len)
    End If
End If
If position<>0 Then
Call exitlong("매수추적스탑",Atstop,hhv(1,high,barnumsinceEntry+1) - _
atr(atrlen)*len1)
Call exitshort("매도추적스탑",Atstop,llv(1,low,barnumsinceEntry+1) + _
atr(atrlen)*len1)
End If

```

〈표 0-17〉 테스트 결과

평가 항목	V0.21	V0.22	V 0.3
총 손익	85,187.11	93,116.18	92,964.53
평균 손익	78.37	97.81	98.90
최대 손실	-1,132.98	-1,132.98	-1,132.98
총 매매수	1,087	952	940
연속 이익 매매수	9	10	9
연속 손실 매매수	9	7	7
최대 자본 인하액	-3,834.51	-4,438.61	-4,382.02
승률	47.65	49.47	49.57
손익비	1.53	1.68	1.69
평균손익비	1.68	1.72	1.72
보상 비율	22.22	20.98	21.22

주) 단위: 천원, %, 배

총손익 기준으로 같은 방향으로 진입이 제한된 v0.22가 가장 좋은 결과를 나타내지만, 최대자본인하액과 보상비율 등의 위험수준과 함께 고려한다면, 매매횟수와 같은 방향 진입제한이 결합된 v0.3이 더욱 안정적이라 할 수 있다.

이제 데이트레이딩 전략 따라하기의 마지막으로 시간대 변경 부분을 알아보기로 하자.

4) 시간대(time frame) 변경

위의 전략을 적용하는 시간대를 변경해보고 다시 최적화를 해봄으로 최적의 매매 시간대 및 전략의 범용성(robust)을 검증해볼 수 있다. 범용성 검증이 가능한 이유는 다양한 시간대에서 수익이 발생 한다면, 어떤 시간대를 사용해도 무관하다는 결론을 얻을 수 있기 때문이다.

테스트는 DD_RangeBreak(v0.3)을 이용해서 3분, 5분, 7분, 10분, 15분, 20분, 25분, 30분 차트를 비교해보기로 하자. 최적화를 진입부터 모든 변수를 실행해서 최적의 결과를 만들 수 있지만, atr기간값(atrlen)과 배수(len1)만을 이

용해서 최적화를 해보기로 하겠다.

테스트 조건은 다음과 같다.

1차 테스트 조건

- ATRlen: 10 ~ 50, 증감치: 10
- Len1: 3 ~ 5, 증감치: 0.2

1차 테스트 결과 15분 이상부터 atrlen 10, len1=1~2 사이가 최적 변수임

2차 테스트 조건

- Atrlen: 5 ~ 15, 증감치: 1
- Len1: 1 ~ 4, 증감치: 0.2

〈표 0-18〉 테스트 결과(DD_RangeBreak(v0.3))

평가 항목	3분	5분	7분	10분
총 손익	93,822.40	92,002.26	93,274.65	92,964.52
평균 손익	98.86	97.36	99.23	98.90
최대 손실	-1,327.50	-1,027.07	-1,132.98	-1,132.98
총 매매수	949	945	940	940
연속 이익 매매수	9	9	9	9
연속 손실 매매수	7	7	6	7
최대 자본 인하액	-4,543.84	-4,166.24	-4,451.95	-4,382.02
승률	50.47	49.31	51.06	49.57
손익비	1.65	1.68	1.67	1.69
평균손익비	1.62	1.73	1.60	1.72
보상 비율	20.65	22.08	20.95	21.22
len	0.37	0.37	0.37	0.37
atr len	100	40	50	20
len1	5.5	3.8	3.4	2.7
tcount	2	2	2	2

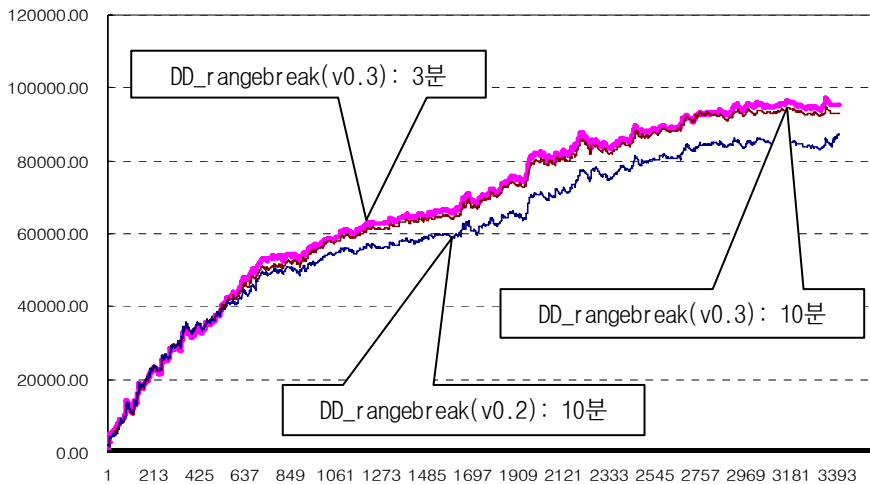
평가 항목	15분	20분	25분	30분
총 손익	83,812.25	79,737.59	81,597.17	83,122.69
평균 손익	89.64	85.92	89.57	90.65
최대 손실	-1,191.43	-1,147.25	-1,155.23	-1,961.54
총 매매수	935	928	911	917
연속 이익 매매수	6	9	8	6
연속 손실 매매수	7	7	7	7
최대 자본 인하액	-5,232.52	-5,173.20	-4,532.73	-3,983.74
승률	48.88	48.71	49.62	49.51
손익비	1.60	1.59	1.58	1.60
평균손익비	1.67	1.67	1.60	1.63
보상 비율	16.02	15.41	18.00	20.87
len	0.37	0.37	0.37	0.37
atr len	14	10	9	8
len1	2.2	1.8	1.8	1.6
tcount	2	2	2	2

주) 단위: 천원, %, 배

<표 3-18>의 테스트 결과 모든 시간대에서 수익이 유지되며 3분에서 가장 높은 수익을 나타냄을 알 수 있다. 또한 시간대가 짧아 질수록 atrlen이 길어지고 배수값(len1)은 커지게 된다. 이러한 현상은 시스템 트레이딩을 개발함에 있어서 시간대 별로 어느 정도의 변수를 사용할 것인가에 대한 기준이 될 수 있을 것이다.

모든 시간대에서 수익이 발생하는 것을 알 수 있어 범용적인(robust) 매매 전략임을 다시 한번 확인할 수 있었다. 그러면 여기서 10분 차트를 이용한 v0.2 와 v 0.3 과 최고 총손익을 보인 3분 차트 v0.3 의 누적 수익곡선을 관찰해보자.

<그림 0-29> 누적 수익곡선



<그림 3-29>를 보면 3분 및 10분 차트에 적용한 v0.3이 v0.2의 수익곡선 보다 꾸준히 높은 수준을 유지하고 있음을 알 수 있다.

따라서 전략 개발 → 진입제어 → 시간대 조절 과정을 통해서 점점 더 수익구조가 좋아졌다는 판단이 가능해진다. 특히 짧은 시간대 적용이 2003년 들어 수익곡선이 극도로 나빠지는 구간에서도 횡보내지 소폭 상승이 가능하다는 것을 보여주고 있다.

이제 단기매매 전략 개발을 먼저 이해 보도록 하자.

3. 단기매매 (Short Term Trading)

단기매매는 short-term trading, swing trading이라 한다. swing trading 이란 의미로 사용되는 경우는 두 가지로 구분되는데 전략 자체가 가격의 스윙 (상승 → 하락 → 상승)을 전략화한 스윙 전략이란 의미가 있다. 이러한 주가의 스윙 중에서 짧은 기간의 스윙(5일 이내)을 단기 매매에서 자주 사용하기 때문에 단기매매를 스윙트레이딩이라 한다.

어쨌든 단기매매는 장 종반에 무조건 청산을 하지는 않는 것이다. 즉 시점에 따라서 오버 나잇으로 수익을 창출하는 방법이다. 앞서 설명한 바와 같이 오버나잇을 할 경우 시가 형성이 현재 포지션에 급격하게 반대로 움직이는 위험(overnight risk)을 감수하는 매매 형태이다. 물론 현재 포지션과 같은 방향으로 시가가 형성되면서 수익이 증가할 수도 있다.

선물매매에서 대부분의 전략들은 오버나잇을 하지 않는 데이트레이딩 전략이다. 하지만 큰 수익의 획득은 오버나잇을 하여 큰 위험을 감수함으로써 가능해진다.

단기매매는 분차트를 이용한 단기매매와 일간차트를 이용한 단기매매를 생각할 수 있다.

(1) 분 차트를 이용한 단기매매

여기서는 앞서 설명한 RangeBreak전략을 분차트에 적용하되 2가지 청산 전략을 추가하기로 한다.

- 일정 시간대에서, 매수일 경우 금일 저가가 매도 라인 이상으로 진행된 적이 있다면 당일 청산, 매도일 경우 금일 고가가 매수라인 이상으로 진행된 적이 있다면 당일 청산을 한다. → 가격이 장 중 양방향으로 진행하였다면 당일청산한다.
- 진입 이후 지정한 날이 지난 시점의 시가와 진입가격을 비교하여 수익이 발생한 상태이면 청산한다(Profitable Open Stop).

조심할 것은 앞서 설명한 데이트레이딩과 마찬가지로 다음과 같은 오류를 제어해야 한다.

- 동시호가에 주문 발생하지 않도록 진입금지를 15 :05 분으로 설정한다.

- 시가에 주문오류가 발생하지 않도록 시간 제어(ttime<1500) 사용해야 한다.
- Atstop주문일 경우 수익 발생 후 바로 재주문이 발생하는 오류를 제어한다.

전략식과 신호식은 다음과 같다.

〈수식 0-21〉 분차트를 이용한 단기매매전략

영역: 전략

이름: SS_RangeBreak(v 0.2)

Input: len(0.37), atrlen(20), len1(3.2), entrystart(920), entrylimit(1430), dayin(3), method(0)

```

'기본식-----
If method = 0 then
Var1=highd(1)-lowd(1)
Cond1= exitname(1)="매수추적스탑" And tdate=exitdate(1) And tdate=Entrydate(1) _
And exitprice(1)>opend+Var1*len
Cond2= exitname(1)="매도추적스탑" And tdate=exitdate(1) And tdate=Entrydate(1) _
And exitprice(1)<opend-Var1*len
Cond3= exitname(1)="매수추적스탑" And tdate=exitdate(1) And tdate>Entrydate(1) _
And exitprice(1)>opend+Var1*len
Cond4= exitname(1)="매도추적스탑" And tdate=exitdate(1) And tdate>Entrydate(1) _
And exitprice(1)<opend-Var1*len
If ttime>Entrystart And ttime<Entrylimit Then
If Cond1=False And Cond3=False Then
Call buy("매수",Atstop,Def,opend+Var1*len)
End If
If Cond2=False And Cond4=False Then
Call sell("매도",Atstop,Def,opend-Var1*len)
End If
End If
If position>0 Then
Call exitlong("매수추적스탑", Atstop, hhv(1,high,barnumsinceEntry+1)- atr(atrlen)*len1)
Call exitshort("매도추적스탑", Atstop, llv(1,low,barnumsinceEntry+1)+atr(atrlen)*len1)
End If
'손실발생 당일청산-----
Elseif method = 1 or method = 2 or method = 3 then
If ttime =1450 And position>0 Then '2시 50분에 포지션이 있다면
If position=1 And Entryprice > close Then '진입가격 보다 종가가 낮다면
Call exitlong("당일청산", Atmarket)
End If

```

```

    If position=-1 And Entryprice < close Then
        Call exitshort("당일청산", Atmarket)
    End If
End If
'방향성 청산-----
Elseif method = 2 or method = 3 then
    If ttime =1450 And position<0 Then
        If position=1 And (opend-Var1*len > lowd Or Entryprice > close) Then
            Call exitlong("당일청산", Atmarket)
        End If
        If position=-1 and ( opend+var1*len < highd or Entryprice < close) Then
            Call exitshort("당일청산", Atmarket)
        End If
    End If
End If
'Profitable Open Stop-----
Elseif method = 3 then
    If position=0 Then '날짜 변경을 카운트하기 위한 식
        Var50=0 ' 초기화
    End If
    If tdate>tdate(1) And position<0 Then '포지션이 있으면서 날짜가 변경되면
        Var50=Var50 +1 '카운트를 1씩 증가시킨다.
    End If
    If Var50=dayin And tdate>tdate(1) And position<0 Then
        ' 카운트가 지정한 숫자가 되면
    End If
    If Entryprice < opend Then ' 진입가격보다 시가가 크면
        Call exitlong("시가청산", Atmarket) '청산
    End If
    If Entryprice > opend Then
        Call exitshort("시가청산", Atmarket)
    End If
End If
End If

```

이 전략식은 기본진입전략과 ATR추적스탑을 기본으로 한 단기매매전략에 3가지 청산전략을 결합한 것이다. 입력변수 중 method를 1로 입력하면 손실발생 당일 청산추가, 2는 방향성 청산추가, 3은 Profitable Open Stop이 추가되어 실행된다.

Cond1은 금일 매수하여 금일 수익을 내고 청산한 후, 바로 재 매수되는 오류를 제어한 것이다. Cond3은 금일 이전에 매수하여 금일 수익을 내고 청산한 후, 바로 재 매수되는 오류를 제어한 것이다. Cond2, Cond4는 매도에러 제어에 해당한다.

기본전략에 3가지 청산전략이 추가되어 성과가 변화하는 과정을 확인하기 위해 단계별로 추가하면서 전략을 적용하였다. 결과는 다음과 같다.

〈표 0-19〉 SS_Rangebreak(v0.2) 단계별 적용결과

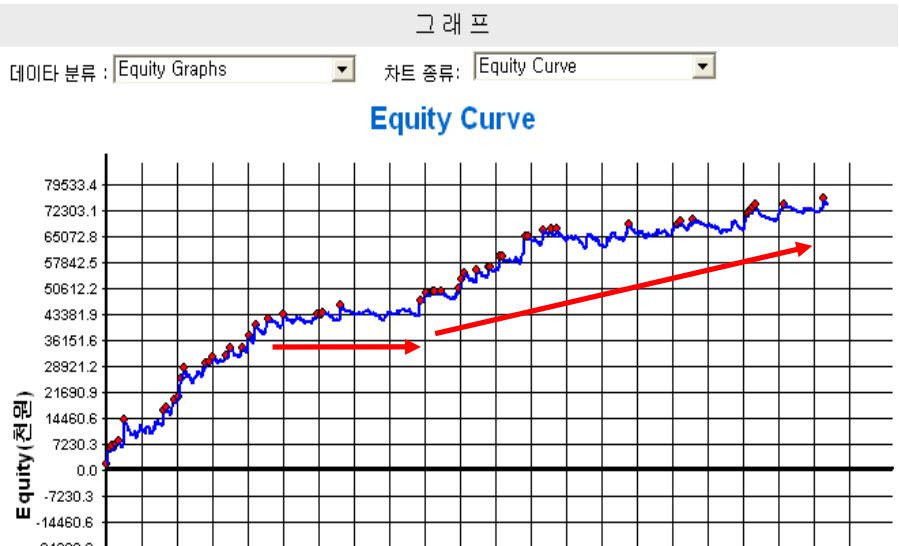
평가 항목	기본식	손실청산추가	방향성청산 추가	시가 청산 추가
총 손익	71,593.48	63,495.11	72,103.35	72,851.43
평균 손익	84.73	73.32	82.22	82.97
최대 손실	-2,565.24	-1,574.60	-1,446.55	-1,446.55
총 매매수	845	866	877	878
연속 이익매매수	9	8	8	8
연속 손실매매수	12	12	9	9
최대자본인하액	-6,993.00	-6,780.04	-5,336.05	-5,336.05
승률	39.88	36.37	39.79	39.98
손익비	1.36	1.35	1.42	1.42
평균손익비	2.05	2.37	2.15	2.14
보상 비율	10.24	9.37	13.51	13.65

주) 단위: 천원, %, 배, 필수적으로 보유해야할 바 수: 50

기본식에 손실청산을 적용하면 최대손실이 대폭 감소하는 현상을 볼 수 있으나, 총손익이 감소하고 승률이 저하된다. 하지만 여기에 방향성 청산을 추가할 경우 총손익이 유지되면서 최대손실, 연속손실매매수가 줄며 보상비율, 손익비 등이 향상되는 것을 알 수 있다.

최종적으로 시가청산전략까지 추가하면 총손익, 최대손실, 최대자본인하액, 승률, 보상비율 등 전략전반의 성과를 향상시킬 수 있다.

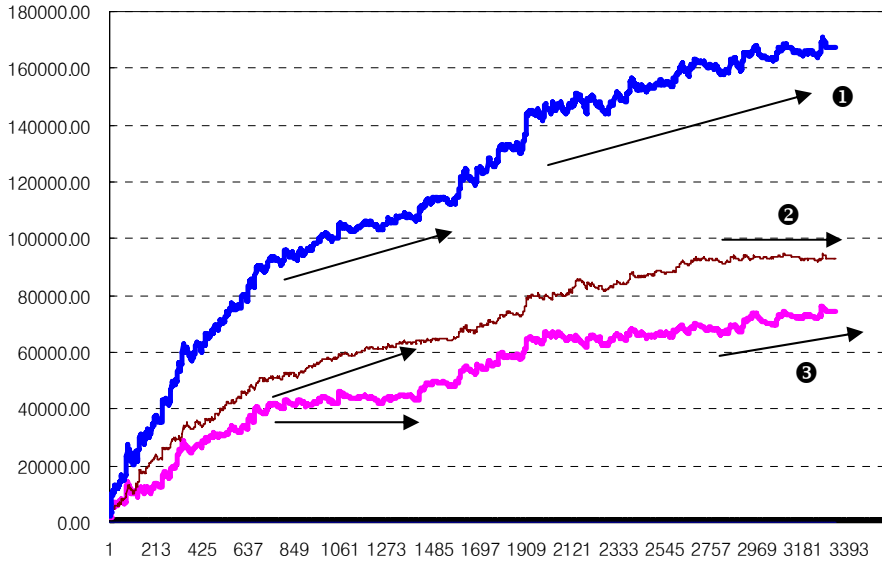
〈그림 0-30〉 SS_Rangebreak(v0.2) 누적수익곡선



여기서 포트폴리오 트레이딩을 생각해보자. 단기매매인 SS_RangeBreak(v0.2)와 데이트레이딩인 DD_RangeBreak(v0.3)을 동시에 매매하면 어떤 효과가 발생할 것인가?

앞서 설명한 데이트레이딩 전략은 최근 국면에서 수익감소가 발생하였고, 단기매매는 수익곡선이 점진적으로 증가되었다. 따라서 두 가지 전략을 동시에 사용하면 현재 국면에서 서로 상쇄, 내지 보완되는 효과를 나타낼 것이다. 다음 그림을 보자.

〈그림 0-31〉 포트폴리오 트레이딩



②는 데이트레이딩 전략인 DD_RangeBreak(v0.3)이다. ③은 단기매매 전략인 SS_RnangeBreak(v0.2)로서, 두 가지 전략을 상호 보완, 상쇄되는 움직임을 보여 결합 수익곡선(①)의 상승 및 선형성이 증가하였음을 알 수 있다. 특히 최근 장세에서도 수익곡선의 상승세를 유지하고 있음을 보여준다.

이와 같이 투자자본이 허용된다면 포트폴리오 전략매매로 위험을 감소시키는 방법이 바람직하다.

그럼 연결선물 지수 일간차트를 이용한 단기매매 전략을 만들어 보자.

(2) 일간차트를 이용한 단기매매

일간 단기매매전략에서 진입방향성을 결정하는 방법이 다양하다.

- 전전일 증가가 전일 증가 보다 낮아지면 매수, 반대면 매도: 역 추세적인 진입 방법으로 증가 하락 이후 반등을 기대하여 매수 방향만 허용하는 방법.
- 전전일 증가가 전일 증가 보다 높아지면 매수, 반대면 매도: 추세적인 진입 방법으로 증가 상승 이후 추세적인 상승 지속을 기대하고 매수 방향으로 매매하는 방법.
- 증가 비교, 고가 비교, 저가 비교, 평균값 비교, 시가 비교 등을 통한 변형
- 기술적 지표로서 방향성 결정: 이동평균선 보다 주가가 낮은 수준이면 매도 방향만, 높은 수준이면 매수 방향으로 매매하는 방법.

여기서는 이러한 방향성 결정 방법 중 역추세적인 진입방법과 ATR추적스탑, 시가청산전략을 혼합하여 전략을 작성하겠다.

일간 연결선물지수에 전략을 적용할 경우 아래의 문제점들을 고려해야 한다.

- `Open(-1)` 사용: `RangeBreak` 전략은 해당일 시가를 이용해서 매매를 한다. 그러나 CT의 주문은 전일봉에서 조건이 만족되어야 발생하므로 해당일의 시가를 이용해서 매매하는 경우 다음날 매매가 발생한다. 그러나 `Open(-1)`을 이용하면 금일봉에서도 주문을 발생시킬 수 있다(`high(-1)`, `low(-1)`, `close(-1)`은 사용 못함). 분차트에서도 동일하다.
- 같은 봉에서 동시 진입 제어: 같은 봉에서 동시 진입(매수, 매도)이 발생할 경우 실시간 매매에서는 가능한 매매 방법이 되나, 과거 테스트에서는 봉이 형성된 과정을 알 수가 없기 때문에 같은 봉 진입은 오류로 간주 된다. 따라서 과거 테스트를 정확하게 하기 위해서는 같은 봉에서 매수, 매도주문이 동시에 발생하는 경우를 제어하는 함수(`setbaroneentry`)를 사용한다.

〈수식 0-22〉 일간차트를 이용한 단기매매 전략

영역: 전략

이름: S_Rangebreak(v 0.2)

Input: len(0.4), atrlen(3), profitlen(2), stoplen(4), len1(21), method(0)

‘진입전략’-----

Var1=high-low ‘시가 고가의 거리 = range

Setbarback ‘ 주문 발생시점 한 봉 전으로

If open(-1)<=open then ‘ 내일 시가가 오늘 시가보다 적거나 같으면

Call buy("매수",Atstop,Def,open(-1)+var1*len)

End if

If open(-1)>open then ‘ 내일 시가가 오늘 시가보다 크면

Call sell("매도",Atstop,Def,open(-1)-var1*len)

End if

If method = 1 then

‘시가청산조건식’-----

If i_position<>i_position(1) Then

Var10=0 ‘초기화

Var11=0 ‘초기화

End If

If i_position=1 And open(-1)>=i_Entryprice + atr(atrlen) Then

‘현재 매수상태이며 진입 이후 시가가 진입가보다 일정수준 이상이면

Var10=var10+1 ‘var10을 1씩 증가

Elseif i_position=-1 And open(-1)<=i_Entryprice - atr(atrlen) Then

‘현재 매도상태이며 진입 이후 시가가 진입가 보다 일정수준 이하이면

Var10=var10+1 ‘var10을 1씩 증가

End If

If i_position=1 And open(-1)<=i_Entryprice Then

‘현재 매수상태이며 진입 이후 시가가 진입가 보다 일정수준 이하이면

Var11=var11+1 ‘var11을 1씩 증가

elseif i_position=-1 And open(-1)>=i_Entryprice Then

‘현재 매도상태이며 진입 이후 시가가 진입가 보다 일정수준 이상이면


```

    Var11=var11+1   'var11을 1씩 증가
End If
'수익발생시가청산-----
If position=1 Then
    If Var10=profitlen Then
        Call exitlong("시가청산",Atmarket)
    End If
Elseif position=-1 Then
    If Var10=profitlen Then
        Call exitshort("시가청산",Atmarket)
    End If
End If
'손실발생시가청산-----
If position=1 Then
    If Var11=stoplen Then
        Call exitlong("손실청산",Atmarket)
    End If
Elseif position=-1 Then
    If Var11=stoplen Then
        Call exitshort("손실청산",Atmarket)
    End If
End If
'ATR추적스탑-----
If position=1 Then
    Call exitlong("추적스탑",Atstop,hhv(1,high,i_barnumsinceEntry+1)-atr(atrlen)*len1)
Elseif position=-1 Then
    Call exitshort("추적스탑",Atstop,llv(1,low,i_barnumsinceEntry+1)+atr(atrlen)*len1)
End If
End If

```

〈수식 0-23〉 일간차트를 이용한 단기매매 신호

영역: 신호

이름: S_Rangebreak(v 0.2)

Input: len(0.4), atrlen(3), len1(2.1)

```

Var1:=high(1)-low(1) '전일시가 고가의 거리 = range
If open<=open(1) then
Call plots1("매수라인",open+var1*len)
End if
If open>open(1) then
Call plots2("매도라인", open-var1*len)
End if
If i_position=1 Then
Call plots3("추적스탑", hhv(1,high, i_barnumsinceEntry+1)-atr(atrlen)*len1)
Elseif i_position=-1 Then
Call plots3("추적스탑", llv(1,low, i_barnumsinceEntry+1)+atr(atrlen)*len1)
End If

```

〈표 0-20〉 S_Rangebreak(v0.2)

평가 항목	기본진입전략	청산전략추가
총 손익	108,087.34	113,694.63
평균 손익	360.29	338.38
최대 손실	-7,378.65	-3,691.87
총 매매수	300	336
연속 이익 매매수	11	8
연속 손실 매매수	9	13
최대 자본 인하액	-12,530.02	-8,106.99
승률	54.00	52.38
손익비	1.87	1.87
평균손익비	1.60	1.70
보상 비율	8.63	14.02

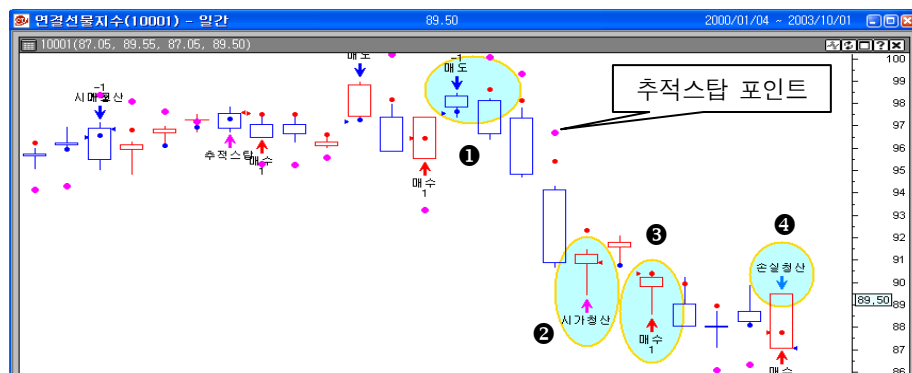
주) 최적 변수: len(0.4), atrlen(3), profitlen(2), stoplen(4), len1(2.1), 필수보유바수: 10

<수식 3-22>에서 수익발생시가청산은 진입 이후 시가를 기준으로 이익이 일정 수준이상 두 번 발생하면 시가에 청산하는 방법이며, 손실발생시가청산은 진입 이후 시가를 기준으로 손실이 일정수준이상 세 번 발생하면 청산하는 방법이다.

<표 3-20>은 적용결과이다. 청산전략을 포함한 전략이 총손익, 최대 손실, 최대자본인하액, 평균손익비, 보상비율 등에서 상대적으로 좋은 성과를 내는 것을 알 수 있다. 이는 수익이 증가될 뿐 아니라 위험도 줄여주는 구조로서 청산전략을 적용하는 것이 보다 효율적인 전략임을 알게 해준다.

신호선을 작성하여 전략과 함께 적용한 그림은 다음과 같다.

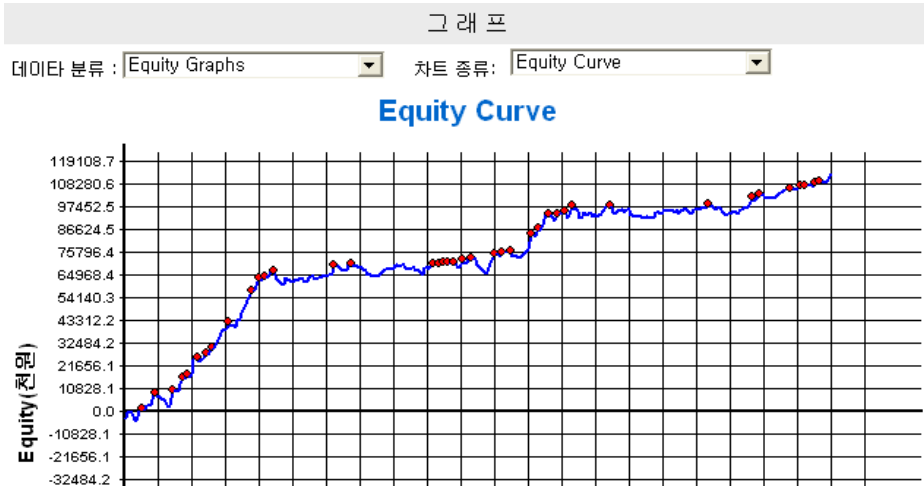
<그림 0-32> 일간차트를 이용한 단기매매 전략



①에서 매도 하여 시가 수익이 두 번 발생한 ②에서 수익발생 시가 청산하였고, ③에서 매수하여 3봉 시가 손실이 발생한 ④에서 손실발생 시가 청산하였다. 그림에 표시된 추적스탑 포인트는 마지노선으로서 상당히 넓은 스탑포인트를 가지고 있음을 알 수 있다. 또한 전략식에서 매수, 매도 방향성 제어를 사용하였으므로 setbaroneentry함수를 사용하지 않았다.

수익곡선은 다음과 같다.

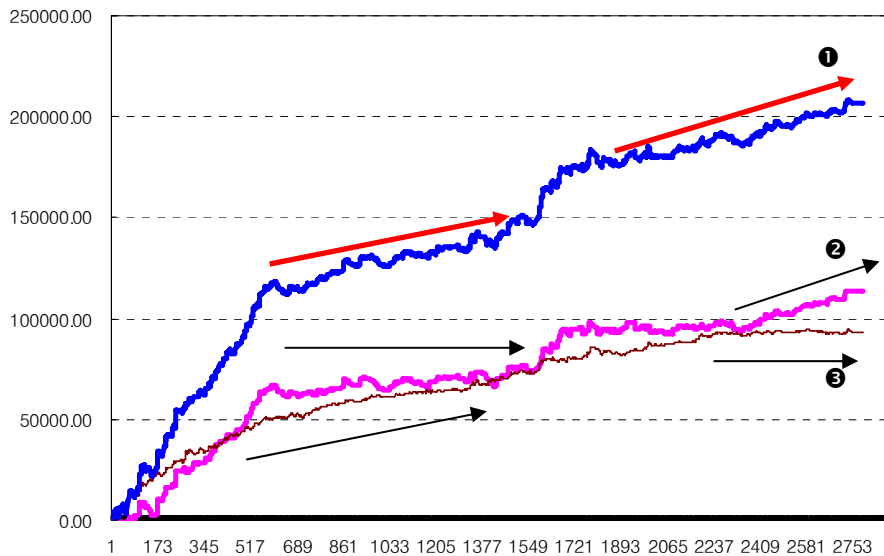
〈그림 0-33〉 수익곡선



역시 현재 시점에서 수익곡선이 상승하는 단기매매(오버나잇)의 효과를 잘 보여주고 있다.

데이트레이딩 전략과 결합하여 포트폴리오 수익곡선의 변화를 보기로 하자.

〈그림 0-34〉 포트폴리오 수익곡선



<그림 3-34>에서 ③이 데이트레이딩 전략인 DD_RangeBreak(v0.3), ②가 일간 단기매매 전략인 S_RangeBreak(v0.2), ①이 결합된 수익곡선이다.

그래프에 화살표들을 보면 두 가지 전략이 서로 상호보완적으로 작용하여 결합수익곡선(①)의 증가 추세가 상당히 개선되는 것을 알 수 있다. 특히 현시점과 같이 추세가 있으면서 변동성이 낮은 국면에서는 단기매매 전략이 유용하다는 것을 알 수 있다. <그림 3-31>와 비교하여도 <그림 3-34>의 결과가 더 좋은 것을 알 수 있다.

시스템 트레이딩 따라하기를 통해서 데이트레이딩 전략, 단기매매 전략 등 두 가지를 작성해 보았다.

4. 정리

다음페이지의 <표 3-21>에 제시된 전략은 총손익 기준 최고의 전략이 아닌, 적절한 수익과 위험 구조를 가진 것들로 선정하였다.

최대 손실을 보면 데이트레이딩(DD v0.3)과 분간 단기매매 전략(SS v0.2)은 유사한 금액이지만 일간 단기매매전략(S v0.2)일 경우에는 226% 증가하여 일간단기매매 전략의 위험이 크다는 것을 보여준다. 최대자본인하액은 데이트레이딩보다 분간 단기매매 전략이 22% 증가하였고 일간 단기매매 전략은 데이트레이딩 전략보다 85% 증가하였다.

수익 측면을 보면, 분간 단기매매의 경우 총손익이 -22%, 평균손익이 -16% 감소하였으며, 일간 단기매매 전략의 경우에는 총손익이 22% 증가, 평균손익이 242%증가로 큰 폭 향상되었다.

수익과 위험의 증가율을 비교해 본 결과, 데이트레이딩 전략에서 분간 단기매매로 전환시 위험은 증가하는 반면 수익은 감소하였다. 일간 단기매매로 전환

할 경우 위험도 증가하고 수익도 증가하는 양상이나, 분간 단기매매보다 위험이 더 커진다는 것을 알 수 있다. 따라서 소규모자본으로 투자하는 일반투자자는 데이트레이딩으로 접근하는 것이 바람직하다는 것을 기억하기 바란다.

〈표 0-21〉 테스트 결과

평가 항목	DD_RangeBreak (v0.3)	SS_RangeBreak (v0.2)	S_RangeBreak (v0.2)
총 손익	92,964.53	72,851.43(-22%)	113,694.63(22%)
평균 손익	98.90	82.97(-16%)	338.38(242%)
최대 손실	-1,132.98	-1,446.55(28%)	-3,691.87(226%)
총 매매수	940	878	336
연속 이익 매매수	9	8	8
연속 손실 매매수	7	9	13
최대 자본 인하액	-4,382.02	-5,336.05(-22%)	-8,106.99 (85%)
승률	49.57	39.98	52.38
손익비	1.69	1.42	1.87
평균손익비	1.72	2.14	1.70
보상 비율	21.22	13.65	14.02
비고	10분 차트 len : 0.37 atrlen : 20 len1 : 2.7 Tcount : 2	10분 차트 len : 0.37 atrlen : 20 len1 : 3.2 Entrystart : 920 Entrylimit : 1430 dayin : 3	일간차트 len : 0.4 atrlen : 3 profitlen : 2 stoplen : 4 len1 : 2.1

주) 괄호 안의 비율은 DD_RangeBreak(v0.3) 기준 증감율임

〈표 0-22〉 제공된 전략들

구분	전략명	설명
데이트레이딩 (Day Trading)	DD_RangeBreak(v0.1)	비율 추적스탑 적용
	DD_RangeBreak(v0.2)	ATR 추적스탑 적용
	DD_RangeBreak(v0.21)	당일 매매 횟수 제한
	DD_RangeBreak(v0.22)	같은 방향 매매 제한
	DD_RangeBreak(v0.3)	매매 횟수 + 같은 방향 매매 제어
	매매 시간대 변경	3, 5, 7, 10, 15 분 등
단기매매 (Short -Term Trading)	SS_RangeBreak(v0.2)	추적스탑 + 방향성 청산 + 손실 시가 청산 + 이익시가 청산
	S_RangeBreak(v0.2)	진입 방향 제어 + 이익시가 청산 + 손실시가 청산 + 추적스탑